

Copy No. 41

## RESEARCH REPORT

ON

## OPTIMAL CONTROL FOR SATELLITE ATTITUDE MANEUVERS

(Contract No. NAS 5-10191 and NAS 5-10349)

APR 1968

## VOLUME 2: COMPUTER PROGRAMS

GPO PRICE \$ \_\_\_\_\_

CFSTI PRICE(S) \$ \_\_\_\_\_

Hard copy (HC) 3.50Microfiche (MF) 1.50

ff 653 July 65

FACILITY FORM 602

(ACCESSION NUMBER)

(THRU)

(PAGES)

(CODE)

(NASA CR OR TMX OR AD NUMBER)

(CATEGORY)

RA750402A



PART VII

DIGITAL AND HYBRID SIMULATOR COMPUTER PROGRAMS

## Preface

One of the principal results of this work was a computer program incorporating the analysis described in the first book of the report. This program contained the control computation in a digital program and the dynamics available either digitally or in an analog computer for hybrid computation. The program writeup contained here was separated from the main body since it was felt that the actual programming and computation instructions would not have as wide an audience.

The description of the analog computer flow and the hybrid setup contained in section 5 - 9 were prepared by Computing and Software, Inc. to whom we express our appreciation for their contribution to the completeness of this report.

The description of the digital program in sections 1 - 4 is, of course, the responsibility of The Mathematical Sciences Group.

PART VII  
TABLE OF CONTENTS

	<u>Page No.</u>
Preface.....	1
7.1 Digital Program Description.....	3
7.2 Verification Information.....	18
7.3 Flow Charts For Digital Program.....	40
7.4 Digital Program Listings.....	52
Ideal Model.....	53
Sensor Model.....	70
Elasticity Model.....	91
7.5 Simulation Model For Analog Computer.....	108
7.6 Overall Hybrid Simulation.....	112
7.7 Simulation Setup Procedures.....	114
7.8 Analog Patching.....	115
Initial Condition Setup.....	116
Dynamics & Control.....	118
Thrustor Misalignments.....	126
Thrustor Real Effects & Fuel Consumption.....	130

## 7.1 *Description of a Computer Program for Optimal Control Synthesis and Satellite Attitude Manuevering Simulation.*

### I. *Subprogram Description*

The program is written in Fortran IV for the SDS 9300; an alternate version is acceptable to the IBM 7094. Following are the names of the various subprograms together with their approximate word length: "MAIN" - 1400; "INIT" - 510; "STUP" - 116; "TOR" - 1403; "REG" - 1130; "DTOA" - 143; "ATOD" - 66; "OUT" - 86; "RK" - 270; "DER" - 222. Operation of the program may be purely digital, or hybrid in conjunction with the HYDAC 2000. In the former mode both the control synthesis and the dynamics are done digitally; in the latter, the attitude simulation is done by the HYDAC 2000, angles and rates being sent to the 9300, it determining applicable controls and transmitting these back to the analog computer.

"MAIN" is the executive branch of the program. It reads all input quantities, determines the mode of operation, calls upon the various subroutines at the proper time, does some computation necessary for initialization and ground track determination and decides when a run is to be terminated and a new run is to begin. Further, in optional versions of the program which include sensor and elastic effects these calculations are also done in "MAIN".

Subroutine "INIT" does the major portion of the initialization prior to actual execution. Input to the program is in terms of

latitude and longitude on the earth, whereas control is with respect to a coordinate system whose origin is in the satellite. "INIT" determines the initial angles and angular rates in this coordinate system.

"STUP" determines the elements of the three by three matrix,  $C$ , referred to in the section on coordinate systems. Three angles are furnished as arguments of the subroutine; the nine matrix elements are computed.

"TOR" determines torques on the three principal axes which are caused by solar pressure and the earth's gravity gradient. It was demonstrated by a series of runs that these torques are sufficiently small so that they may be replaced by small constant torques without essential change in either angles or rates during the course of a single slewing maneuver. Since this subroutine requires over 1400 storages, as various additions and modifications were made to the program it was found necessary to replace it by a dummy routine in order to comply with storage restrictions on the 9300.

"REG" incorporates all normalization and logic for the determination of control. After all angles and rates have been sampled "REG" is called to determine for each of the three axes whether the control torque should be positive, negative or zero, considering the weighting of the cost function parameters. "REG" then further considers the estimated arrival time of all three axes assuming these control policies are adopted. It then modifies these policies to

prevent large differences in arrival times while guaranteeing that the maximum overall time to reach the target is not increased. Controls are then transmitted to "MAIN".

"DTOA" is a digital-analog interface subroutine which transmits three controls and three disturbance torques to the analog computer each time the subroutine is called upon by the timed interrupt.

"ATOD" is a similar subroutine to "DTOA", except that here three angles and three angular rates are transmitted from the analog to the digital computer.

"OUT" is the output subroutine which does all the printing.

"RK" does the numerical integration of the six differential equations via the usual fourth order Runge Kutta formulas. When elastic effects are included in the program ten differential equations are integrated.

"DER" merely calculates the values of the six or ten derivatives involved in the dynamics. It is called upon by "RK" four times for each complete forward step.

## II. *Input to Ideal Model Program*

Card 1: FORMAT (2I5, 5F10.5)

Field 1: NA serves two purposes. If zero the hybrid computer is to be used, control being handled digitally while the dynamics are simulated on the analog. If positive the program is to be executed digitally only, in which case NA is the number of integration steps to be taken between samplings for possible control changes.

Field 2: NB is the number of samplings between printouts. Thus if NB is 1 output will be printed at each sample time; if NB is 90000 no output will occur in a normal run.

Fields 3-5: PA, PB and PC are all scale factors which are used in transmitting torques, angles and angular rates digital to analog so as to keep the voltages on the analog computer within the range of the equipment. These factors depend naturally upon the wiring of the analog patch board. The most recently used factors were  $-5.0 \times 10^3$ , 5.0 and 250.0 .

Field 6: Z is merely a dummy variable to indicate whether the disturbance torques are to be calculated from the subroutine "TOR" or whether they are to be read in as constants on card 6. If Z equals 0 the subroutine is to be used, if not the torques are to be read in.



## Card 2: FORMAT (9F8.5)

Fields 1-3: ZI(1-3) are the moments of inertia about the three principal axes, the latest values being 3580, 1970 and 2000 slugs/ft<sup>2</sup>.

Fields 4-6: ZM(1-3) are the thrusts exerted by the jets on the three principal axes, the latest values being .08, .04 and .12 lb. respectively.

Field 7: AL(1) is a multiplier of the cross-coupling terms in the differential equations and gives the user the capability of removing these terms from the digital dynamics by setting it to zero. For the complete dynamics AL(1) = 1; for single axis dynamics AL(1) = 0.

Field 8: AL(2) is unused and should be unpunched.

Field 9: AL(3) is the longitude of the satellite, i.e., the satellite sub-point, with the convention that a negative quantity represents a west longitude and a positive an east longitude. Since the satellite has usually been assumed to be over the United States, -100. has been most commonly used.

## Card 3: FORMAT (9F8.5)

Fields 1-9: ((ZL(I,J), J = 1,3), I = 1,3) are the weighting parameters which determine the cost function to be minimized. The index I denotes the axis, the index J refers to the parameter being weighted, i.e., time, fuel or energy. Although the capability for independent weighting on different axes exists, no application has been made of this feature.

Thus, fields 4-6 and 7-9 are repetitions of 1-3. For a run to be strictly time optimal card 3 should read 1, 0, 0, 1, 0, 0, 1, 0, 0. Such a run, for a Mohave-Quito maneuver, takes 131 seconds. The same run with card 3 reading 1, 2, .00005, 1, 2, .00005, 1, 2, .00005 takes 178 seconds and uses about one-half the fuel. With card 3 reading 1, 8, .00005, 1, 8, .00005, 1, 8, .00005 the run takes 306 seconds and uses about one-third the fuel.

While it is true that the weighting parameters for a cost function such as this should sum to one, for computational purposes it is not necessary here. In order to study the effects of varying these parameters it is more feasible to fix two of them and vary a third. It is reasonable to set  $ZL(1,J)$  equal to 1 for all cases. If one wishes to study variations of  $ZL(2,J)$  then fixing  $ZL(3,J)$  at .00005 will permit any positive values for  $ZL(2,J)$ . However, values larger than 10 will result in no significant fuel savings. Similarly, to study  $ZL(3,J)$  setting  $ZL(2,J)$  at .00005 permits all positive values for  $ZL(3,J)$ . In this case the maximum rate is limited by approximately

$$\sqrt{\frac{1}{ZL(3,J)}} \cdot$$

Card 4: FORMAT (9F8.5)

Field 1: AX is used in time synchronization, the first step of which is to find which of the three angles will require the longest time to reach the origin. The estimated time for this angle is then multiplied by AX and this product called TMAX. TMAX is now compared to the es-

timated time for each of the other angles to determine whether control policy should be changed to make arrival times more nearly the same.

Field 2: BX is a safety factor for the time synchronization. The estimated time of arrival for any axis whose thruster is to be cut off for time synchronization purposes will be at least BX seconds less than TMAX.

Field 3: XF was originally used to prevent zig-zagging on the final leg of the path to the origin. 10. is a good value for this parameter.

Field 4-6: PIB, ROB, YIB are the angles (in radians) of misalignment of the "B" system with respect to the "P" system and are normally assumed to be zero.

Card 5: FORMAT (9F8.5)

Field 1: YO is the angle of rotation about the pointing axis 2, in degrees, when the body is in the holding mode. This angle must always be very small and is usually input as zero.

Fields 2-3: AT and ONG are the initially assumed latitude and longitude in degrees to which the satellite is pointing when the maneuver is to begin.

Fields 4-5: XLAT and XLON are the latitudes and longitudes of the target in degrees.

Field 6: DT is the time increment (in seconds) which is used as the step size in the integration subroutine.

Field 7: AE was an experimental parameter which is now fixed at 5. Its purpose is to prevent chattering along the rate limiting line  $\omega = S$ . It sets a minimum band width for which thrusters are to be cut off. Its units are radians per second. However it must be multiplied by the ratio of moment of inertia to thrust before being input. Since this ratio is approximately 20,000, the value 5 actually represents a band width of about .00025 radians/sec.

Card 6: FORMAT (9F8.5) - Optional

Fields 1-3: T(1), T(2), T(3) are the disturbance torques in foot-pounds which are used to replace the calculated torques from TOR when Z of card 1 is nonzero. If Z is zero this card should not appear.

Card 7: FORMAT (9F8.5)

Fields 1-3: Elevation misalignment angles for the positive axes 1-3.

Fields 4-6: Azimuth misalignment angles for the positive axes 1-3.

All angles are in degrees.

Card 8: FORMAT (9F8.5)

Fields 1-6: This card is the counterpart of card 7 for negative axes.



### III. *Input for Sensors*

The input for the operation of the program with sensor real effects differs only in card 4 from that of the ideal model. Fields 1-3 are AX, BX and XF as before. Fields 4 and 5 are ZK1 and ZK2, which correspond to parameters K1 and K2 in the sensor write-up of the technical report, and are currently .1 and .005. Field 6 is SD, and is used in the random number generator to determine the standard deviation currently 0.00063. Fields 7-9 are PIB, ROB and YB, which in the ideal model input appeared in Fields 4-6.

The values of ZK1, ZK2, and SD are not arbitrary but are very much determined by satellite system specifications. The value for SD has been carefully derived on the basis of oral communications with the appropriate GSFC technical persons. That derivation is given in Appendix F, Part 1 (Filter specification (choice of K)). The values of ZK1 and ZK2 are intimately involved with the signal noise SD, the sampling interval  $\delta$  and also the modeling errors. This dependence is described also in Appendix F, Part 1 (computation of K.) Referring to Appendix F, Part 1, Fig. 3, we can see the region from which possible (not necessarily good) values of K may be selected.

### IV. *Input for Elastic Effects*

Card input here is identical to that of the ideal model through card 5. Card 6 uses fields 1-3 for the three disturbance

torques as before. Field 4 is for the length of the boom in feet. Field 5 is for the mass of the boom in slugs. Field 6 is for the spring constant in  $\text{slugs/sec}^2$  and field 7 the damping coefficient, which is dimensionless. It should be noted that the moment of inertia of the elastic boom is already included in the moments of inertia about the principal axes.

#### V. *Operating Instructions*

Several sense switch options are available to the operator. When running digitally sense switch 1 may be set, whereupon computations will halt and a new print frequency, NB, is to be typed in FORMAT (15) on the on-line typewriter. Before depressing the carriage return switch 1 should be reset to prevent halting again. The print frequency may be changed any number of times during a run using this procedure.

Switch 2 is used for terminating a run and reading in new data. When switch 2 is set computations will cease, and terminal conditions printed if in digital mode. After inserting new data into the card reader it is only necessary to push the carriage return to start a new run.

When operating in the hybrid mode when the satellite has reached its target control will be cut off, and no communication is possible between HYDAC and the 9300 until at least one plotter pen has been removed from the origin and then switch 3 set. Hybrid operation will resume at this time.

Setting sense switch 4 allows one to type in YO, AT, ONG, XLAT and XLON in FORMAT (5F5.3) after all data cards have been read in. These typed values will override the corresponding values already read in. However, this must take place at the beginning of a run before internal looping has begun.

In all digital runs when the target is reached terminal conditions will be printed on the on-line printer and "PAUSE 16" will appear on the typewriter. Inserting a new set of data and pushing the carriage return will start a new run. In hybrid runs after initial conditions are printed computations will cease and "SET ANALOG - PAUSE 11" will be typed. After setting up the HYDAC the run is initiated by the carriage return.

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

[REDACTED]

\_\_\_\_\_

[illegible][illegible][illegible][illegible][illegible][illegible]

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000 1001 1002 1003 1004 1005 1006 1007 1008 1009 1010 1011 1012 1013 1014 1015 1016 1017 1018 1019 1020 1021 1022 1023 1024 1025 1026 1027 1028 1029 1030 1031 1032 1033 1034 1035 1036 1037 1038 1039 104

[illegible]

~~~~~

[illegible][illegible]

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80





## OUTPUT

Input from cards 7 and 8 are printed first in the same order. Two  $3 \times 3$  matrices are then printed, these being multipliers corresponding to positive and negative control torques. If we refer to the elements of the first matrix as  $A_{ij1}$  and those of the second as  $A_{ij2}$ , then let

$$b_{ij} = \begin{cases} A_{ij1} & \text{if } U_i \geq 0 \\ A_{ij2} & \text{if } U_i < 0 \end{cases},$$

where  $U_i$  is the current control torque for axis  $i$ . The effective control torque  $\bar{U}_i$  is then given by

$$\bar{U}_i = \sum_{j=1}^3 b_{ij} U_j$$

Following a page eject the other input parameters appear as follows.

- Line 1: All weighting factors  $((ZL(I,J), J=1,3), I=1,3)$   
 Line 2: Yaw angle, initial latitude and longitude, target latitude and longitude, angles  $\rho, Y$  and  $\pi$  in the BS system, DT.  
 Line 3: AX, BX XF, AE.  
 Line 4: T(1-3), followed by the scaled torques for HYDAC.

The body of the run appears next, and for the ideal model its format follows.

- Line 1: Time,  $\rho, Y, \pi, \omega_1, \omega_2, \omega_3$ , latitude and longitude of the pointing vector.

Line 2:  $\rho, Y, \pi, \omega_1, \omega_2, \omega_3$ .

Line 3: Control torques  $(u_1, u_2, u_3), T(3)$ , normalized  $\rho$ , normalized  $\omega_1, S$ .

The body of a run with sensors has identical lines 1, 2, and 3 as that of an ideal model run. An additional line appears, however, which contains filtered angles  $\rho, Y, \pi$ , estimated  $\omega_1, \omega_2, \omega_3$  and unfiltered noisy angles  $\rho, Y$  and  $\pi$ .

The body of an elastic run has the following appearance.

Line 1: Time,  $\rho, Y, \pi, \omega_1, \omega_2, \omega_3, q_1$  and  $q_3$ , (displacements of the point mass with respect to axes 1 and 3),  $\dot{q}_1$  and  $\dot{q}_3$ .

Line 2:  $\dot{\rho}, \dot{Y}, \dot{\pi}, \dot{\omega}_1, \dot{\omega}_2, \dot{\omega}_3, \dot{q}_1, \dot{q}_1, \ddot{q}_3$  and  $\ddot{q}_3$ .

Line 3: XM, XL, D(13)-spring constant, D(14)-damping coefficient.

Line 4:  $u_1, u_2, u_3$ , ground track latitude and longitude.

## 7.2 Verification Information

### 7.2.1 Introduction.

An integral part of the construction of any digital program is the checkout. This involves establishing that the formulae which represent the analysis have been correctly mechanized and also verifying that the analysis is true, perhaps by showing that it works for simple cases such as, in our program, single axis slews.

A great variety of experience with the program, only a portion of which is described in the two parts of this report, has given us great confidence that it simulates the motion correctly. Part of this confidence stems from the modular construction, enabling us to use established sub-routines in many places.

### 7.2.2 Euler Angle Calculations.

An elementary test for initial condition calculation involves the single-axis relations. That is, the computation of the Euler angles  $(\phi_{br}, \psi_{br}, \theta_{br})$  of the body in the target system when only one angle is perturbed. This procedure works, for instance, when the satellite points to its subpoint with no yaw and the target lies in the equatorial plane,  $\theta_{br} \neq 0$ .

Figure 7.2.1 shows

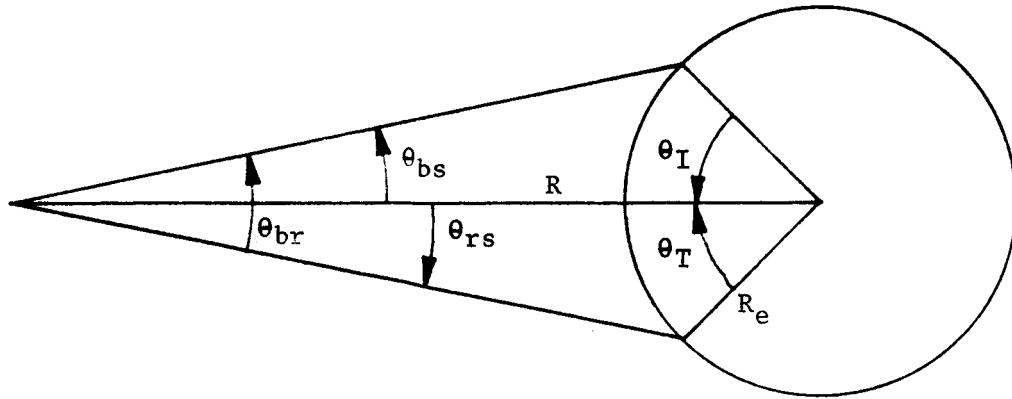
$$\theta_{br} = \tan^{-1} \frac{\sin \theta_T}{K - \cos \theta_T} - \tan^{-1} \frac{\sin \theta_I}{K - \cos \theta_I}$$

where  $\theta_T$  is target longitude, positive east of Greenwich.

$\theta_I$  is initial longitude, positive east of Greenwich.

$K = 6.610808$ , the ratio of the orbit radius of a synchronous satellite to the radius of the earth,  $= \frac{R}{R_e}$ .





$$\theta(\theta_T, \theta_I) = \theta(\theta_T, 0) - \theta(\theta_I, 0),$$

and

$$\theta(-\theta_T, 0) = -\theta(\theta_T, 0).$$

Therefore, only the basic curve  $\theta(\theta_T, 0)$   $0 \leq \theta_T \leq 90$  is really necessary.

For latitude calculations, note that

$$\phi(\phi_T, \phi_I) = -\theta(\theta_T, \theta_I).$$

For check purposes, several points on the curves were computed by the machine and marked with x's, the calculations being done both in latitude and longitude.

A more detailed analysis for three axis freedom was then performed. It checks explicitly only the calculation of the initial euler angles  $(\phi, \psi, \theta)_{br}$  as determined by target latitude and longitude, initial latitude and longitude, initial "yaw" angle  $y$ , and the longitude of the satellite subpoint. Actually, however, these checks are rather more

comprehensive, since the calculations are performed in a subroutine which is used elsewhere in the program.

The following procedure was followed in checking the transformations.

The pertinent formulae were derived by persons not involved in the original problem definition. These formulae were programmed on a small digital computer using a non-Fortran language and not referring to the code presently used in the simulator.

Then a number of spot checks were run and the results compared with those from the simulator. These results appear in Table 7.2.1.

The formulae used in the small computer are:

$$\tan \theta_{rs} = - \frac{\cos \phi_T \sin \theta_T}{K - \cos \phi_T \cos \theta_T}$$

$$\tan \phi_{rs} = \frac{\sin \phi_T}{\cos \theta_{rs} (K - \cos \phi_T \cos \theta_T) - \cos \phi_T \sin \theta_T \sin \theta_{rs}}$$

$\phi_T$  = target latitude

$\theta_T$  = target longitude

$K$  = 6.610808 ratio of synchronous orbit radius to earth radius.

$\theta_{rs}$ ,  $\phi_{rs}$  are the pitch and roll angles through which the S system must be rotated, pitch ( $\theta_{rs}$ ) first, to align with the R (target) system.

$$\tan \theta_{Is} = - \frac{\cos \phi_I \sin \theta_I}{K - \cos \phi_I \cos \theta_I}$$

$$\tan \phi_{Is} = \frac{\sin \phi_I}{\cos \theta_{Is} (K - \cos \phi_I \cos \theta_I) - \cos \phi_I \sin \theta_I \sin \theta_{Is}}$$

$\phi_I$  = Initial latitude

$\theta_I$  = Initial longitude.

$\theta_{Is}$ ,  $\phi_{Is}$  are the pitch and roll angles through which the S system must be rotated to point at the initial latitude and longitude and maintain minimum angle between the body 3-axis and north.

We now assume an arbitrary rotation of  $-y$  takes place about the body 2-axis. This rotates the three axis out of a plane defined by the satellite center and a north pointing vector through the target, and thus increases the angle between the polaris sensor and north.

This will not be the body yaw because yaw is the second euler angle. This is the yaw angle of a second set of euler angles.

Using this yaw angle along with  $\theta_{Is}$  and  $\phi_{Is}$  we can define a new set of angles  $\theta_{bs}$ ,  $\psi_{bs}$ , and  $\phi_{bs}$  which are the euler angles of the body with respect to the S system including the yaw angle  $y$ .

These are given by:

$$\begin{aligned}\psi_{bs} &= \sin^{-1}(\sin y \cos \phi_{Is}) \\ \phi_{bs} &= \sin^{-1}(\sin \phi_{Is} / \cos \psi_{bs}) \\ \theta_{bs} &= \sin^{-1}((\sin \theta_{Is} \cos y - \sin y \sin \phi_{Is} \cos \theta_{Is}) / \cos \psi_{bs}).\end{aligned}$$

Having the euler angles of the body with respect to the S system and of the R system with respect to the S system, we can solve for the euler angles of the body with respect to the R-system.

These are:

$$\begin{aligned}\psi_{br} &= \sin^{-1}(\sin \phi_{rs} \cos \psi_{bs} \sin(\theta_{rs} - \theta_{bs}) + \cos \phi_{rs} \sin \psi_{bs}) \\ \theta_{br} &= \text{sgn}(\theta_{bs} - \theta_{rs}) \cos^{-1}((\cos \psi_{bs} \cos(\theta_{rs} - \theta_{bs})) / \cos \psi_{br}) \\ \phi_{br} &= \sin^{-1}((\cos \phi_{rs} \sin \phi_{bs} \cos \psi_{bs} - \sin \phi_{rs} \cos \phi_{bs} \cos(\theta_{rs} - \theta_{bs}) \\ &\quad - \sin \phi_{rs} \sin \phi_{bs} \sin \psi_{bs} \sin(\theta_{rs} - \theta_{bs})) / \cos \psi_{br})\end{aligned}$$

Theoretical investigations show that the largest yaw ( $y_{br}$ ) which can occur "naturally" (initial yaw zero) is in magnitude less than  $2.3^\circ$ . Notice that the largest value appearing here is about  $0.8^\circ$  and occurs on the Quito-Mojave maneuver.

Since both of these calculations were carried out on digital computers, it is proper to ask why even small discrepancies appear. A possible source is the different computers and different input routines. However, we feel that the primary cause is a poorly chosen formula used in the checks - i.e., the simulator is probably more accurate. (Use of inverse cosine in the vicinity of  $\theta = 0$ .)



| Init<br>Lat<br>(deg) | Init<br>Long<br>(deg) | Init<br>Yaw<br>(deg) | Target<br>Lat<br>(deg) | Target<br>Long<br>(deg) | Calc<br>$\phi$<br>(rad) | Calc<br>$\psi$<br>(rad) | Calc<br>$\theta$<br>(rad) | Simulator<br>$\phi$<br>(rad) | Simulator<br>$\psi$<br>(rad) | Simulator<br>$\theta$<br>(rad) |
|----------------------|-----------------------|----------------------|------------------------|-------------------------|-------------------------|-------------------------|---------------------------|------------------------------|------------------------------|--------------------------------|
| 20                   | 30                    | 1                    | 5                      | -10                     | .04337                  | .01916                  | -.11233                   | .04337                       | .01916                       | -.11233                        |
| 20                   | 30                    | 5                    | 5                      | -10                     | .04353                  | .08891                  | -.11537                   | .04354                       | .08891                       | -.11536                        |
| 20                   | 30                    | 10                   | 5                      | -10                     | .04404                  | .17609                  | -.11922                   | .04405                       | .17609                       | -.11922                        |
| 30                   | 10                    | 30                   | -20                    | 15                      | .16878                  | .51842                  | -.06761                   | .16876                       | .51841                       | -.06761                        |
| -30                  | 10                    | 30                   | 20                     | 15                      | -.16858                 | .51645                  | .10047                    | -.16857                      | .51644                       | .10046                         |
| -30                  | -30                   | 30                   | 20                     | 15                      | -.16560                 | .51069                  | .19868                    | -.16559                      | .51068                       | .19867                         |
| -.2                  | -78                   | 0                    | 35.3                   | -1.17                   | -.09884                 | -.01470                 | .14800                    | -.09882                      | -.01470                      | .14800                         |
| -.2                  | -78                   | 0                    | 35.3                   | -117                    | -.08259                 | -.00392                 | .04765                    | -.08258                      | -.00392                      | .04761                         |
| 35.3                 | -117                  | 0                    | -.2                    | -78                     | .08268                  | -.00003                 | -.04781                   | .08268                       | -.00003                      | -.04780                        |
| 35.3                 | -116.9                | 0                    | -.2                    | -78                     | .09932                  | .00007                  | .10649                    | .09932                       | .00007                       | .10648                         |
| 35.3                 | -116.9                | 0                    | -.2                    | -78                     | .08270                  | -.00003                 | -.04770                   | .08270                       | -.00003                      | -.04770                        |
| 0                    | 0                     | -10                  | 80                     | 0                       | -.15412                 | -.17251                 | -.02666                   | -.15412                      | -.17250                      | -.02666                        |
| 0                    | 0                     | 10                   | 80                     | 0                       | -.15412                 | .17251                  | .02666                    | -.15412                      | .17250                       | .02666                         |
| 0                    | 80                    | 10                   | 0                      | 0                       | 0                       | .17453                  | -.15181                   | 0                            | .17453                       | -.15181                        |
| 0                    | -80                   | 10                   | 0                      | 0                       | 0                       | .17453                  | .15181                    | 0                            | .17453                       | .15181                         |

TABLE 7.2.1.

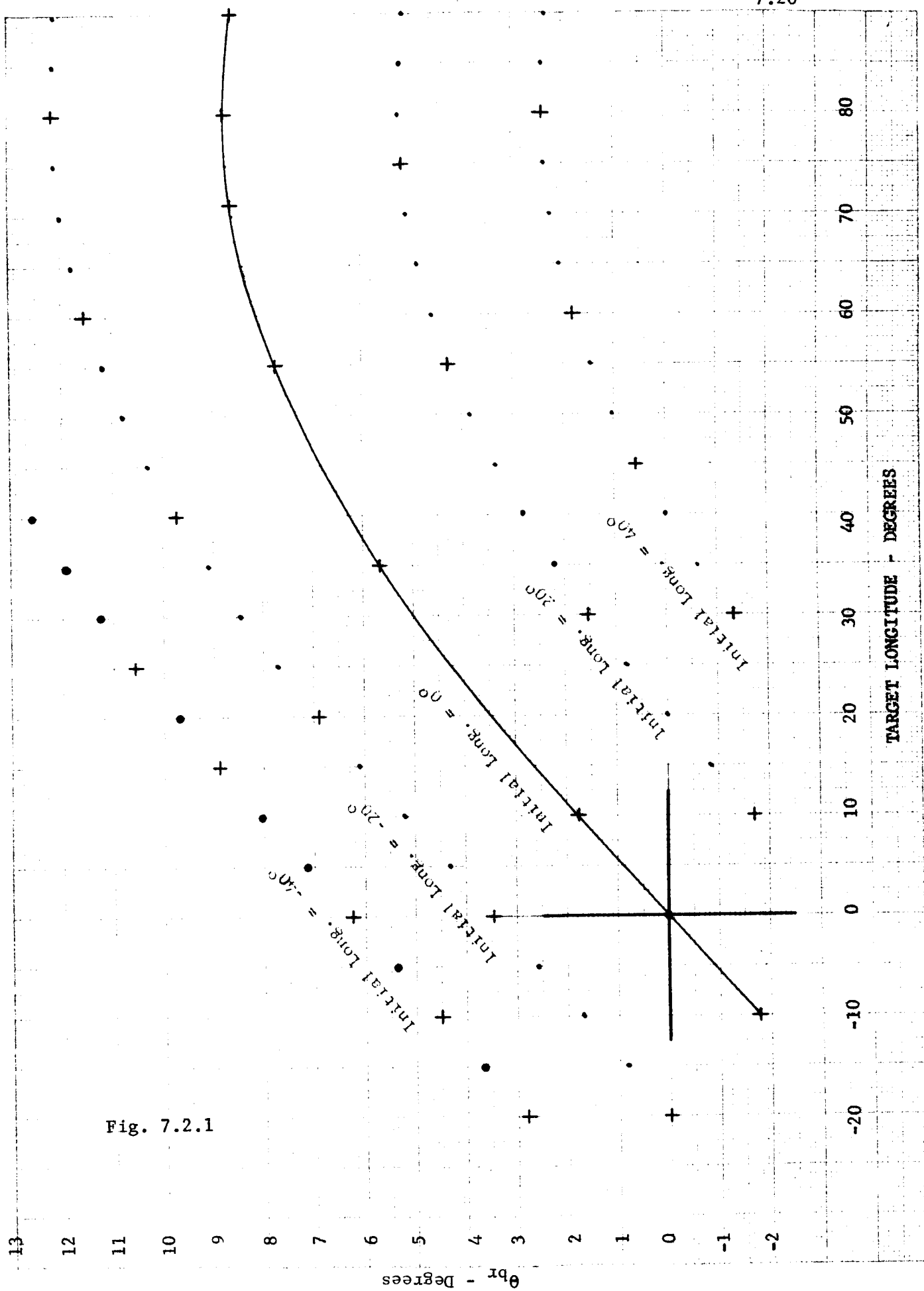


Fig. 7.2.1

Then

$$I_o = \hat{g}^T I \hat{g}$$

which agrees with the equations 2) and 3) in Appendix C.

To check this work further the transformations between the MSG and (3652) systems were substituted directly into (C2) and the same results were obtained in the MSG system.

## II. Torque Calculation.

From (3652) we have

$$T_b^T \cdot B = - [U_\theta, U_\psi, U_\phi], \text{ so}$$

$$T_b = - \begin{bmatrix} \sin \psi & \cos \psi & \sin \phi & \cos \psi & \cos \phi \\ 0 & -\cos \phi & \sin \phi & & \\ 1 & 0 & 0 & & \end{bmatrix}^{-1} \begin{bmatrix} U_\theta \\ U_\psi \\ U_\phi \end{bmatrix}$$

or

$$T_b = - \begin{bmatrix} 0 & 0 & 1 \\ \frac{\sin \phi}{\cos \psi} & -\cos \phi & -\frac{\sin \phi \sin \psi}{\cos \psi} \\ \frac{\cos \phi}{\cos \psi} & \sin \phi & -\frac{\sin \psi \cos \phi}{\cos \psi} \end{bmatrix} \begin{bmatrix} U_\theta \\ U_\psi \\ U_\phi \end{bmatrix}$$

This checks equation 11) of the Appendix C, compared with (C7) of (3652).

$$\begin{aligned} U_\theta &= 2I_{11}a_1(\cos \theta \cos \psi) + 2I_{22}a_2(-\cos \theta \sin \phi \sin \psi \\ &\quad - \sin \theta \cos \phi) + 2I_{33}a_3(\sin \theta \sin \phi - \cos \theta \cos \phi \sin \psi) \end{aligned}$$

### 7.2.3 Verification of Gravity Gradient Torques

Our first task is to verify that the correct equations are used. To do this we compare NASA TN D-3652 with Appendix C.

#### I. Moment of inertia Calculation

On page 33 of 3652 appears the equation

$$I_o = I_{xx}a_1^2 + I_{yy}a_2^2 + I_{zz}a_3^2 + 2I_{xy}a_1a_2 + 2I_{xz}a_1a_3 + 2I_{yz}a_2a_3, \quad (C2)$$

which represents the moment of inertia about the local (geocentric) vertical in terms of the body system moments of inertia.

The vector  $[0, 1, 0]$  is the local vertical unit vector in the MSG S-system.

Referring to Appendix G we find that the local vertical unit vector in the MSG body system will be

$$\hat{g} = \begin{bmatrix} \sin \theta_{bs} \cos \psi_{bs} \\ -\sin \theta_{bs} \sin \phi_{bs} \sin \psi_{bs} + \cos \theta_{bs} \cos \phi_{bs} \\ -\cos \theta_{bs} \sin \phi_{bs} - \sin \theta_{bs} \sin \psi_{bs} \cos \phi_{bs} \end{bmatrix}$$

$$\begin{aligned}
& + 2I_{12}(a_1(-\cos\theta \sin\phi \sin\psi - \sin\theta \cos\phi) + a_2(\cos\theta \cos\psi)) \\
& + 2I_{13}(a_1(-\cos\theta \cos\phi \sin\psi + \sin\theta \sin\phi) + a_3(\cos\theta \cos\psi)) \\
& + 2I_{23}(a_2(-\cos\theta \cos\phi \sin\psi + \sin\theta \sin\phi) + a_3(-\cos\theta \sin\phi \sin\psi - \sin\theta \cos\phi))
\end{aligned}$$

$$\begin{aligned}
U_\psi &= 2I_{11}a_1(-\sin\theta \sin\psi) + 2I_{22}a_2(-\sin\theta \sin\phi \cos\psi) \\
& + 2I_{33}a_3(-\sin\theta \cos\phi \cos\psi) + 2I_{12}(a_1(-\sin\theta \sin\phi \cos\psi) \\
& + a_2(-\sin\theta \sin\psi)) + 2I_{13}(a_1(-\sin\theta \cos\phi \cos\psi) \\
& + a_3(-\sin\theta \sin\psi) + 2I_{23}(a_2(-\sin\theta \cos\phi \cos\psi) + a_3(-\sin\theta \sin\phi \cos\psi))
\end{aligned}$$

$$\begin{aligned}
U_\phi &= 2I_{22}a_2(-\sin\theta \cos\phi \sin\psi - \cos\theta \sin\phi) \\
& + 2I_{33}a_3(\sin\theta \sin\phi \sin\psi - \cos\theta \cos\phi) \\
& + 2I_{12}a_1(-\sin\theta \cos\phi \sin\psi - \cos\theta \sin\phi) \\
& + 2I_{13}a_1(\sin\theta \sin\phi \sin\psi - \cos\theta \cos\phi) \\
& + 2I_{23}(a_2(\sin\theta \sin\phi \sin\psi - \cos\theta \cos\phi) \\
& + a_3(-\sin\theta \cos\phi \sin\psi - \cos\theta \sin\phi))
\end{aligned}$$

$$T_{b1} = \frac{3\mu}{r^3} [-I_{22}a_2a_3 + I_{33}a_2a_3 - I_{12}a_1a_3 + I_{13}a_1a_2 + I_{23}(a_2^2 - a_3^2)]$$

$$T_{b2} = \frac{3\mu}{r^3} [I_{11}a_1a_3 - I_{33}a_1a_3 + I_{12}a_2a_3 + I_{13}(a_3^2 - a_1^2) + I_{23}(-a_1a_2)]$$

$$\begin{aligned}
T_{b3} &= \frac{3\mu}{r^3} [I_{11}(-a_1a_2) + I_{22}(a_1a_2) + I_{12}(a_1^2 - a_2^2) + I_{13}(-a_2a_3) \\
& + I_{23}(a_1a_3)]
\end{aligned}$$

This checks the torque equations. Remember, however, that

$(\theta, \psi, \phi)$  appearing in I) and II) are  $(\theta_{bs}, \psi_{bs}, \phi_{bs})$ , whereas the running angles of the simulation are  $(\theta_{br}, \psi_{br}, \phi_{br})$ .

### III. Numerical Checks.

Having established that the equations are correct, we must verify the programming. To this end a program was written on our terminal and the results compared with the simulator.

An arbitrary case was selected with

$$\phi_{br} = -.072248$$

$$\psi_{br} = .17572$$

$$\theta_{br} = -.029535$$

$$\phi_{rs} = .064267$$

$$\theta_{rs} = .036348$$

This gave

$$\phi_{bs} = -.0353552$$

$$\psi_{bs} = .17453$$

$$\theta_{bs} = .0283034$$

Using a principal axis body system with moments of inertia

$$I_{11} = 2686$$

$$I_{22} = 1846$$

$$I_{33} = 1617 \quad ,$$

both programs gave the same gravitational torques

$$T_1 = -1.10678 \text{ E-7}$$

$$T_2 = 1.44113 \text{ E-8}$$

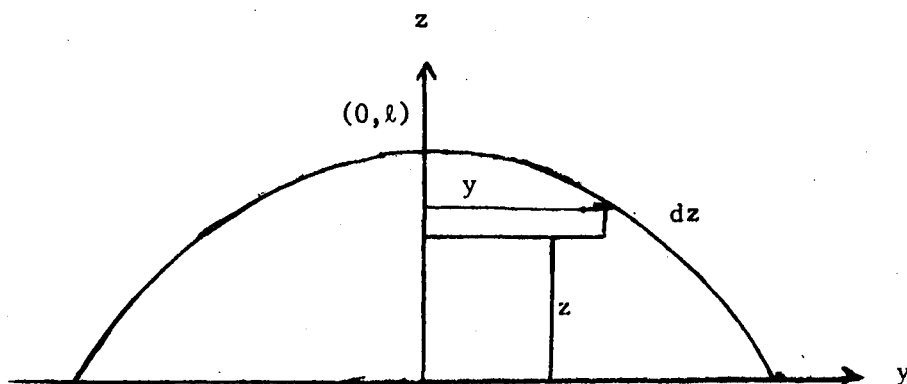
$$T_3 = -3.71911 \text{ E-7} \quad .$$

#### 7.2.4 Verification of Solar Pressure Perturbations

Our object is to review the analysis appearing in Appendix B and to establish that the mechanization of that analysis is correct.

The physical assumptions are fairly standard. The behavior of the photons is considered to be that of small particles with momentum determined by the frequency. Thus momentum is transferred by absorption or by reflection from a smooth surface. The approximation of the satellite by a paraboloid is a necessary approximation to bring the geometry of the problem within reason.

Each equation in the absorptivity section was checked and changes made where necessary (mostly minor typographical changes). For a check on the magnitude and sign of the torque, the absorptivity torque was calculated for the radiation beam coming in at  $90^\circ$  ( $\delta = 90^\circ$ ) to the axis of symmetry of the paraboloid as follows. Since the force due to absorbed photons is the same for any two bodies casting the same shadow, we may calculate the torque using a parabola in the  $yz$  plane.



The equation of the parabola is  $z = \ell - \gamma y^2$ . The element of force is  $dF = (2ydz)S$ , where  $S$  is the solar constant. The element of torque about the  $y$  axis is  $dT = zdF$  and is positive (since  $x$ , the incoming beam, and the force are all out of the page). Other torques about the origin are zero by symmetry. Thus:

$$\begin{aligned} T_a &= 2S \int_0^\ell yzdz \\ &= 2S \int_0^\ell \sqrt{\frac{\ell-z}{\gamma}} z dz \end{aligned}$$

changing variable to  $u = (\ell-z)/\gamma$ , we integrate to

$$T_a = 2S \left[ -\frac{2}{3} \gamma \ell \left( \frac{\ell-z}{\gamma} \right)^{3/2} + \frac{2}{5} \gamma^2 \left( \frac{\ell-z}{\gamma} \right)^{5/2} \right]_0^\ell$$

or

$$T_a = S \sqrt{\frac{\ell}{\gamma}} \frac{8}{15} \ell^2 \quad \text{for } \delta = 90^\circ.$$

Taking the limit as  $\delta \rightarrow 90^\circ$  for the last equation in  $T_a$  on p. 161 of the MSG report, we obtain the same result. For the actual dimensions of the satellite and  $S = 9.4 \times 10^{-8}$  pounds/ft<sup>2</sup>, this is approximately  $T_a = 7.9 \times 10^{-6}$  foot-pounds.

The sign of any absorptivity torque is always positive or zero. If the parabola in the last figure is revolved about the  $z$  axis we have the given paraboloid. The  $x$  axis is out of the page and the  $\hat{b}$  vector (unit vector from the sun to the satellite) makes an acute angle with the positive  $x$  axis and  $y$  is normal to  $\hat{b}$  (definition of  $x$  and  $y$  axes). Since the force is in the direction of  $\hat{b}$ , the torque about the  $y$  axis is always positive or zero. The torque is zero when either the upper or lower surface of the paraboloid is fully illuminated by symmetry of the shadow on the  $z = 0$  (or  $z = \ell$ ) plane. The torque becomes zero when the sun's rays are just tangent to the lower lip



of the paraboloid. Referring to the preceding figure, we may evaluate  $\frac{dz}{dy}$  at the point  $(y,z) = (-\sqrt{\ell/\gamma}, 0)$ . At that point  $\left.\frac{dz}{dy}\right|_{z=0} = 2\sqrt{\ell\gamma}$ . The angle  $\delta_{crit}$  is between the  $z$  axis and the incoming beam so that

$$\tan \delta_{crit} = \frac{1}{2\sqrt{\ell\gamma}}.$$

Another critical angle is obtained from the point  $(+\sqrt{\ell/\gamma}, 0)$  equal to  $\pi - \delta_{crit}$ .

Thus  $T_a$  is non-zero only for

$$\delta_{crit} < \delta < \pi - \delta_{crit}.$$

To check the fact that non-zero  $T_a$  are positive, note in Appendix B, sect. 3 that all terms in  $T_a$  are positive with the possible exception of  $\tan^2 \delta \ell^2 8/15 - \ell/\gamma 4/15$ . Using the smallest value for  $\tan^2 \delta = \tan^2 \delta_{crit}$  and the actual dimensions of the satellite, we see that these terms are also positive.

Each equation in the reflectivity section was checked. There are four cases occurring in the reflectivity situation depending upon the relation of  $\delta$  to  $\delta_{crit}$ . These cases may be summarized.

| Case | $\delta$ range                         | <u>Illuminated</u><br><u>from</u> | <u>Illum. of</u><br><u>Upper Surface</u> | <u>Illum. of</u><br><u>Lower Surface</u> |
|------|----------------------------------------|-----------------------------------|------------------------------------------|------------------------------------------|
| I    | $(0, \delta_{crit})$                   | Below                             | None                                     | Complete                                 |
| II   | $(\delta_{crit}, \frac{\pi}{2})$       | Below                             | Partial                                  | Partial                                  |
| III  | $(\frac{\pi}{2}, \pi - \delta_{crit})$ | Above                             | Partial                                  | None                                     |
| IV   | $(\pi - \delta_{crit}, \pi)$           | Above                             | Complete                                 | None                                     |

To determine the magnitude of this torque, we will determine the maximum value and the signs for case I. For this case, we integrate the equation in Appendix B, section 5 over even limits  $x_0 = -\sqrt{\ell/\gamma}$ ,  $x_1 = +\sqrt{\ell/\gamma}$ . Since the limits are even, on expanding  $(2\gamma x \sin \delta + \cos \delta)^2$

we see that only the cross term  $4\gamma x \sin\delta \cos\delta$  makes any contribution. Differentiating

$$T_r = (-4S)(2\gamma \sin 2\delta) \int_{-\sqrt{\ell/\gamma}}^{+\sqrt{\ell/\gamma}} \int_0^{\sqrt{\ell/\gamma - x^2}} \frac{x[2\gamma^2 x^3 - x(2\gamma\ell - 1) + 2\gamma^2 xy^2]}{1 + 4\gamma^2(x^2 + y^2)} dx dy$$

where we have substituted  $2\sin\delta \cos\delta = \sin 2\delta$  and  $\text{sign}(\cos\theta) = -1$  (for illumination from below, case I).

The integral in the last equation is over half of the base circle.

We may transform to polar coordinates  $r^2 = x^2 + y^2$ ,  $x = r \cos\theta$  to obtain:

$$T_r = (-4S)(2\gamma \sin 2\delta) \int_0^{\sqrt{\ell/\gamma}} \int_0^\pi \frac{(s^2 \cos^2 \theta)[2\gamma^2 r^2 - (2\gamma\ell - 1)]}{1 + 4\gamma^2 r^2} r dr d\theta$$

Since  $1 - 2\gamma\ell > 0$  for this paraboloid, the sign of  $T_r$  is always negative (for case I). The maximum value of  $T_r$  with respect to  $\delta$  is at  $\delta = 45^\circ$  (allowed for case I since  $45^\circ \leq \delta_{\text{crit}} \approx 66.6^\circ$ ). Thus for case I,  $T_r$  is a sine function with amplitude determined by the last integral above.

This last integral may be integrated explicitly. Integrating  $\int_0^\pi \cos^2 \theta d\theta$  we obtain  $\frac{\pi}{2}$ . Changing variable to  $u = 1 + 4\gamma^2 r^2$ , we have

$$T_r = -4S\pi\gamma \sin 2\delta \int_1^{1+4\gamma\ell} \left[ 2\gamma^2 \left( \frac{u-1}{4\gamma^2} \right)^2 + (1-2\gamma\ell) \left( \frac{u-1}{4\gamma^2} \right) \right] \frac{du/8\gamma^2}{u}$$

or

$$\begin{aligned} T_r &= -\frac{1}{2} S\pi \frac{1}{\gamma} \sin 2\delta \int_1^{1+4\gamma\ell} \left[ \frac{1}{8\gamma^2} \left( \frac{1-2u+u^2}{u} \right) + \left( \frac{1-2\gamma\ell}{4\gamma^2} \right) \left( -\frac{1}{u} + 1 \right) \right] du \\ &= -\frac{1}{2} S\pi \frac{1}{\gamma} \sin 2\delta \left[ \frac{1}{8\gamma^2} \left( \ln u - 2u + \frac{u^2}{2} \right) + \left( \frac{1-2\gamma\ell}{4\gamma^2} \right) (-\ln u + u) \right]_1^{1+4\gamma\ell} \end{aligned}$$

32

$$T_r = -\frac{1}{2} S\pi \frac{1}{\gamma} \sin 2\delta \left[ \frac{1}{8\gamma^2} (\ln(1+4\gamma\ell) - 2(4\gamma\ell) + \frac{1}{2}(1+4\gamma\ell)^2 - \frac{1}{2}) \right. \\ \left. + \left( \frac{1-2\gamma\ell}{4\gamma^2} \right) (-\ln(1+4\gamma\ell) + 4\gamma\ell) \right]$$

$$\text{Thus } |T_r|_{\max} = \frac{S\pi}{2\gamma} \left[ \frac{\ln(1+4\gamma\ell) - 4\gamma\ell + 8\gamma^2\ell^2}{8\gamma^2} + \left( \frac{1-2\gamma\ell}{4\gamma^2} \right) (-\ln(1+4\gamma\ell) + 4\gamma\ell) \right]$$

or

$$|T_r|_{\max} = \frac{S\pi}{16\gamma^3} [(-1+4\gamma\ell)\ln(1+4\gamma\ell) + 4\gamma\ell - 8\gamma^2\ell^2] .$$

For this paraboloid:

$$|T_r|_{\max} = 1.83 \times 10^{-4} \text{ ft-pounds.}$$

Since  $\gamma\ell \approx 4.696 \times 10^{-2}$  is small compared to one, we can get an interesting approximation to  $|T_r|_{\max}$ . Expanding the  $\ln$  term and retaining  $\gamma^2\ell^2$  powers, we obtain

$$|T_r|_{\max} \approx \frac{S\pi\ell^2}{\gamma} = S\pi(\sqrt{\ell/\gamma})^2\ell .$$

Now  $\sqrt{\ell/\gamma}$  is the radius of the base circle for the paraboloid and  $\ell$  is the depth. Thus

$$|T_r|_{\max} \approx SV_c$$

where  $V_c$  is the volume of the right circular cylinder which circumscribes the paraboloid and has the same axis of symmetry. This approximation gives

$$|T_r|_{\max} \approx 2.16 \times 10^{-4} \text{ ft-pounds.}$$

The value of  $T_r$  integral for case IV is the negative of that for case I. Since  $\sin 2\delta$  is also negative for case IV, we have that  $T_r$  for

case IV is always negative and that the same value of  $|T_r|_{\max}$  is valid. Since  $|T_r|$  for case I is decreasing as we approach the case II region (and similarly for IV and III), the above value of  $|T_r|_{\max}$  appears to be the maximum for all  $\delta$ .

*Numerical checks.*

In order to check the numerical results for the solar torques computed by the simulator program, several things were done. First, the absorptivity torque ( $T_a$ ) and the reflectivity torque ( $T_r$ ) were calculated separately; second, certain formulas (described below) were re-programmed for the remote console at MSG and the results compared with those of the simulator program; third, the checks previously discussed in this write-up were applied to the numerical results.

The equation used on the remote console for  $T_a$  is:

$$T_a = S \cos^2 \delta \sqrt{\frac{l}{\gamma} - \frac{1}{4\gamma^2 \tan^2 \delta}} \times$$

$$\left[ \frac{l^2}{15} \tan^2 \delta - \frac{l}{\gamma} \frac{4}{15} + \frac{1}{30\gamma^2} \frac{1}{\tan^2 \delta} \right]$$

The four cases occurring for reflectivity are treated according to Table 7.2.4.1 where the functions therein are defined as follows:

$$w(x_0, x_1, y_0, y_1, \delta) \equiv \int_{x_0}^{x_1} (2\gamma x \sin \delta + \cos \delta)^2 [h(x, y_1) - h(x, y_0)] dx$$

where  $x_0$ ,  $x_1$ ,  $\delta$  are constants,  $y_0$ ,  $y_1$  are functions and

$$h(x, y(x)) \equiv \frac{2\gamma^2 x^3 - x(2\gamma\ell - 1)}{4\gamma^2} \left[ \frac{1}{\sqrt{\frac{1+4\gamma^2 x^2}{4\gamma^2}}} \tan^{-1} \frac{y}{\sqrt{\frac{1+4\gamma^2 x^2}{4\gamma^2}}} \right] \\ + \frac{x}{2} \left[ y - \sqrt{\frac{1+4\gamma^2 x^2}{4\gamma^2}} \tan^{-1} \frac{y}{\sqrt{\frac{1+4\gamma^2 x^2}{4\gamma^2}}} \right]$$

The angle  $\alpha$  between the plane of the satellite's orbit and the plane of the earth's orbit was arbitrarily set at  $20.5^\circ$  and a selected set of data appears in Table 7.2.4.2. The angle  $\theta_{se}$  in the table is equal to the angle of the satellite as it travels around the earth in its orbit ( $\theta_{se} = 0^\circ$  being the point closest to the sun). The relation between  $\theta_{se}$  and  $\delta$  for this case is  $\cos\delta = -\cos 20.5^\circ \cos \theta_{se}$ .

We see from Table 7.2.4.2 that there is basic agreement between the two sets of results. The maximum percentage error between the two runs is about 3%. The differences are probably due to the different machines and different integration subroutines used.

For other checks mentioned in this write-up, note that the absorptivity torque  $T_a$  at  $\delta = 90^\circ$  ( $\theta_{se} = 90^\circ$ ) is  $T_a \approx 7.9 \times 10^{-6}$  ft-pounds. We see that the non-zero  $T_a$  is always positive and that  $T_a$  is zero outside of the range

$$\delta_{crit} = 66.6 \leq \delta \leq 113.4^\circ \\ (64.9^\circ \leq \theta_{se} \leq 115.1^\circ) \quad .$$

The reflectivity torque maximum approximately checks  $|T_r|_{max} = 1.83 \times 10^{-4}$  and occurs near  $\delta = 45^\circ$  ( $\theta_{se} = 41.4^\circ$ ). The  $T_r$  torques are always negative for

cases I ( $0 < \delta < 66.6^\circ$ ) and IV ( $113.4 < \delta < 180^\circ$ ) i.e., I ( $115.1 < \theta_{se} < 180^\circ$ ) and IV ( $0 < \theta_{se} < 64.9^\circ$ ).

Figure 7.2.4.1 is a plot of total torque  $T = 0.75 T_r + 0.25 T_a$  versus  $\theta = 180^\circ - \delta$ . In the ranges  $0 < \theta < 66.6$  and  $115.1 < \theta < 180^\circ$ ,  $T_a$  is zero and  $T = 0.75 T_r$ . Thus we see that the  $T_r$  curve in these ranges is indeed a constant times the sine function  $\sin 2\delta$ .

It appears that a very good way to enter the solar torques in the future would be by means of an approximating curve.

From reviewing the tabulated data and from applying the other checks mentioned in the write-up, it appears that the equations in the MSG report are correct and have been correctly programmed.

IV

III

II

I

 $\delta\epsilon(113.425^\circ, 180^\circ]$  $\delta\epsilon(90^\circ, 113.425^\circ]$  $\delta\epsilon(66.575^\circ, 90^\circ]$  $\delta\epsilon(0, 66.575^\circ]$ 

| $T_r$  | $-4Sw(x_o, x_1, y_o, y_1, \delta)$ | $\begin{bmatrix} +4Sw(x_o, x_1, y_o, y_1, \delta) \\ -4Sw(x'_o, x'_1, y'_o, y'_1, \delta) \end{bmatrix}$ | $+4Sw(x_o, x_1, y_o, y_1, \delta)$ | $+4Sw(x_o, x_1, y_o, y_1, \delta)$ |
|--------|------------------------------------|----------------------------------------------------------------------------------------------------------|------------------------------------|------------------------------------|
| $y_o$  | $\equiv 0$                         | $\equiv 0$                                                                                               | $\equiv 0$                         | $\equiv 0$                         |
| $y_1$  | $\sqrt{\frac{\ell}{\gamma} - x^2}$ | $\sqrt{\frac{\ell}{\gamma} - x^2}$                                                                       | $\sqrt{\frac{\ell}{\gamma} - x^2}$ | $\sqrt{\frac{\ell}{\gamma} - x^2}$ |
| $x_o$  | $-\sqrt{\frac{\ell}{\gamma}}$      | $-\sqrt{\frac{\ell}{\gamma}}$                                                                            | $-\sqrt{\frac{\ell}{\gamma}}$      | $-\sqrt{\frac{\ell}{\gamma}}$      |
| $x_1$  | $+\sqrt{\frac{\ell}{\gamma}}$      | $-\frac{1}{2\gamma \tan \delta}$                                                                         | $-\frac{1}{2\gamma \tan \delta}$   | $+\sqrt{\frac{\ell}{\gamma}}$      |
| $x'_o$ | -                                  | $-\frac{1}{2\gamma \tan \delta}$                                                                         | -                                  | -                                  |
| $x'_1$ | -                                  | $+\sqrt{\frac{\ell}{\gamma}}$                                                                            | -                                  | -                                  |
| $y'_o$ | -                                  | $\sqrt{\frac{\ell}{\gamma} - (x + \frac{1}{\gamma \tan \delta})^2}^*$                                    | -                                  | -                                  |
| $y'_1$ | -                                  | $\sqrt{\frac{\ell}{\gamma} - x^2}$                                                                       | -                                  | -                                  |

\*set to zero if quantity  
under square root is  
negative

Table 7.2.4.1

| $\theta_{se}$ (deg) | $T_a$ (ft.-lb.)       |                       | $T_r$ (ft.-lb.)        |                        |
|---------------------|-----------------------|-----------------------|------------------------|------------------------|
|                     | Remote Console        | Simulator Program     | Remote Console         | Simulator Program      |
| 0                   |                       | 0                     | $-1.22 \times 10^{-4}$ | $-1.20 \times 10^{-4}$ |
| 10                  |                       | 0                     | $-1.33 \times 10^{-4}$ | $-1.30 \times 10^{-4}$ |
| 25                  |                       | 0                     |                        | $-1.64 \times 10^{-4}$ |
| 30                  |                       | 0                     | $-1.77 \times 10^{-4}$ | $-1.73 \times 10^{-4}$ |
| 40                  |                       | 0                     |                        | $-1.82 \times 10^{-4}$ |
| 45                  |                       | 0                     |                        | $-1.81 \times 10^{-4}$ |
| 50                  |                       | 0                     |                        | $-1.76 \times 10^{-4}$ |
| 70                  | $6.60 \times 10^{-7}$ | $6.79 \times 10^{-7}$ | $-1.13 \times 10^{-4}$ | $-1.11 \times 10^{-4}$ |
| 80                  | $5.18 \times 10^{-6}$ | $5.23 \times 10^{-6}$ | $-6.58 \times 10^{-5}$ | $-6.50 \times 10^{-5}$ |
| 85                  | $7.16 \times 10^{-6}$ | $7.20 \times 10^{-6}$ | $-4.47 \times 10^{-5}$ | $-4.42 \times 10^{-5}$ |
| 90                  | $7.90 \times 10^{-6}$ | $7.94 \times 10^{-6}$ | $-2.73 \times 10^{-5}$ | $-2.70 \times 10^{-5}$ |
| 100                 | $5.18 \times 10^{-6}$ | $5.23 \times 10^{-6}$ | $-7.15 \times 10^{-5}$ | $-7.07 \times 10^{-5}$ |
| 110                 | $6.60 \times 10^{-7}$ | $6.79 \times 10^{-7}$ | $-1.13 \times 10^{-4}$ | $-1.12 \times 10^{-4}$ |
| 130                 |                       | 0                     |                        | $-1.76 \times 10^{-4}$ |
| 135                 |                       | 0                     |                        | $-1.81 \times 10^{-4}$ |
| 140                 |                       | 0                     |                        | $-1.82 \times 10^{-4}$ |
| 150                 |                       | 0                     | $-1.77 \times 10^{-4}$ | $-1.73 \times 10^{-4}$ |
| 170                 |                       | 0                     | $-1.33 \times 10^{-4}$ | $-1.30 \times 10^{-4}$ |
| 175                 |                       | 0                     |                        | $-1.22 \times 10^{-4}$ |
| 180                 |                       | 0                     | $-1.22 \times 10^{-4}$ | $-1.20 \times 10^{-4}$ |

Table 7.2.4.2



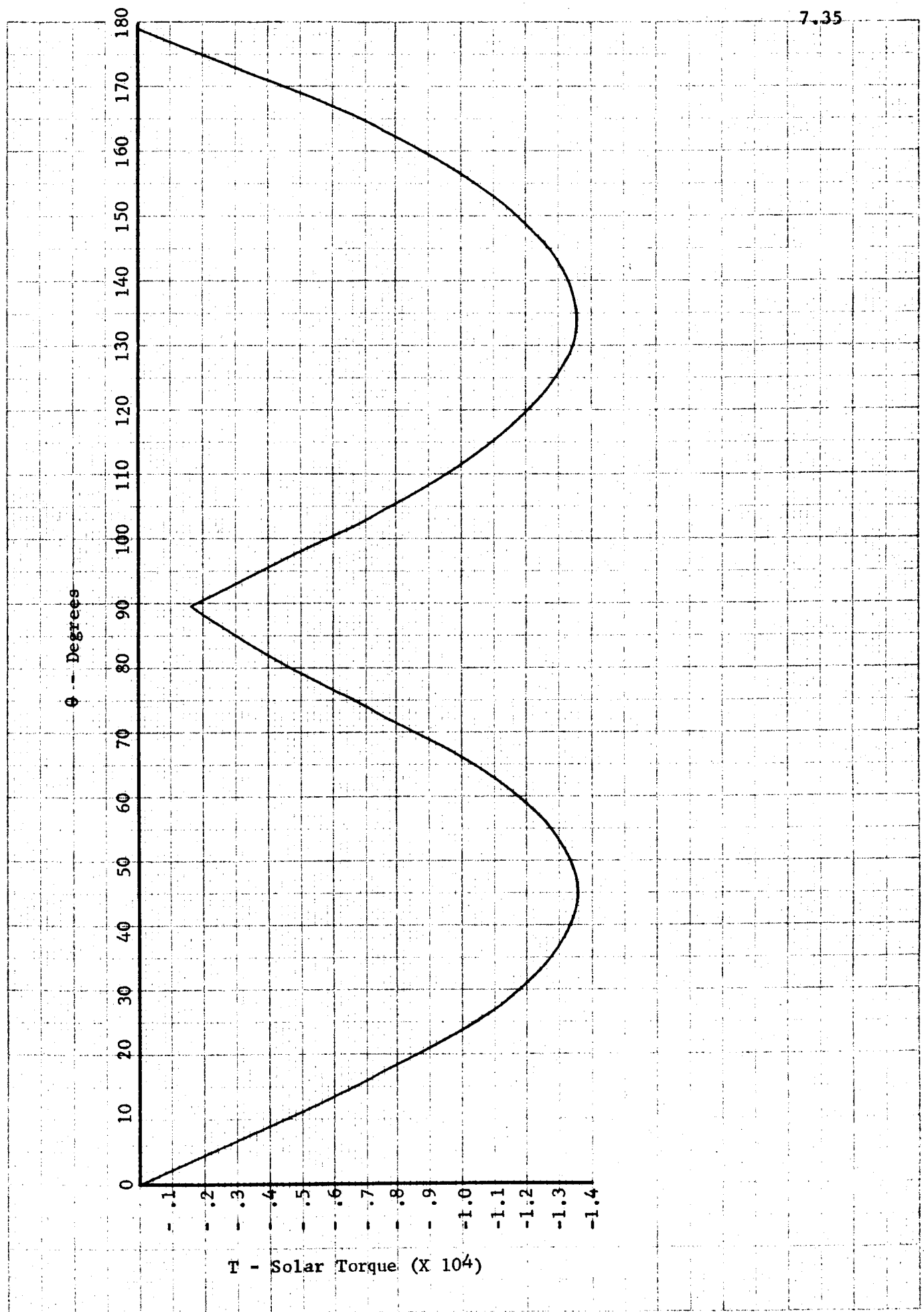
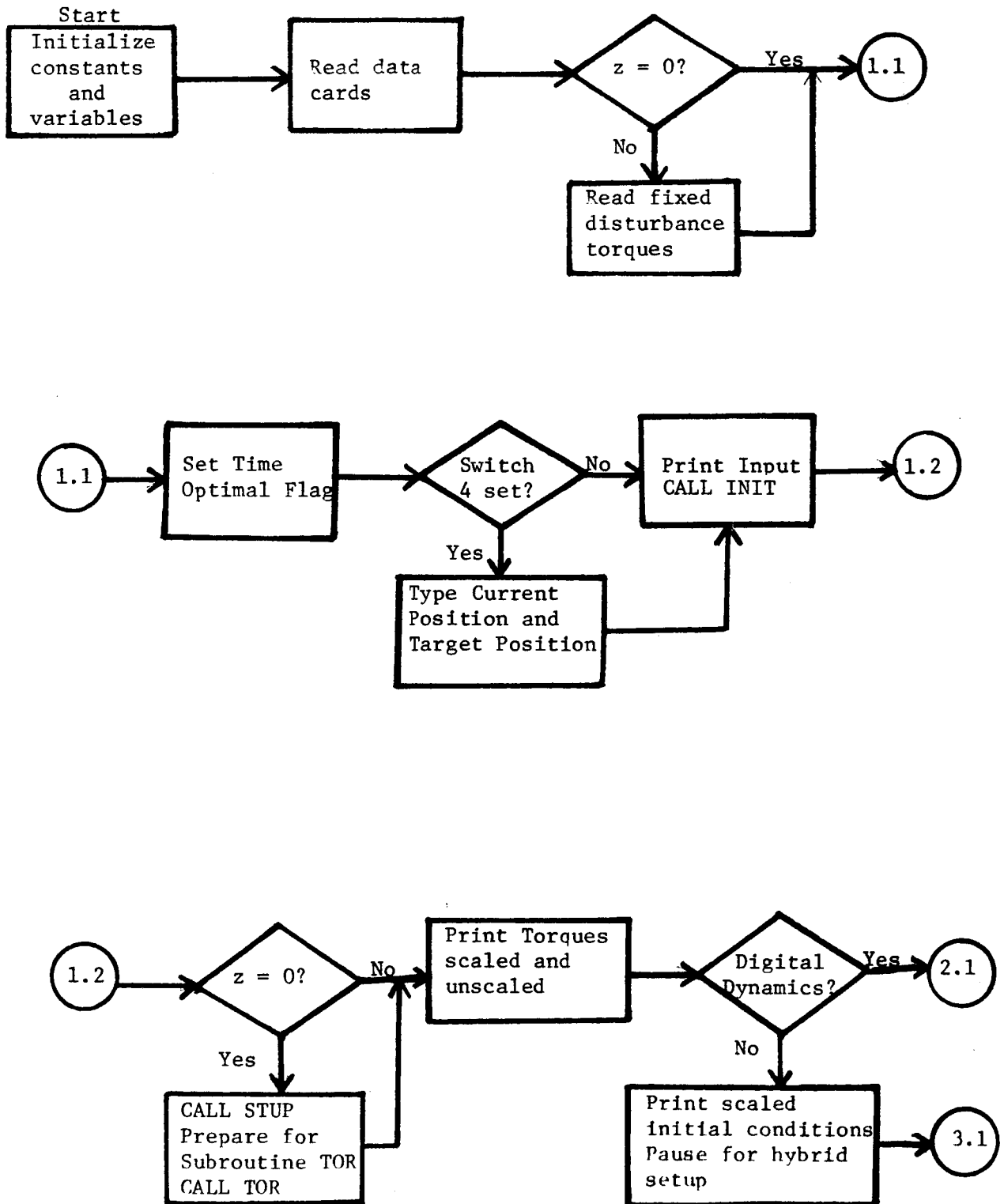
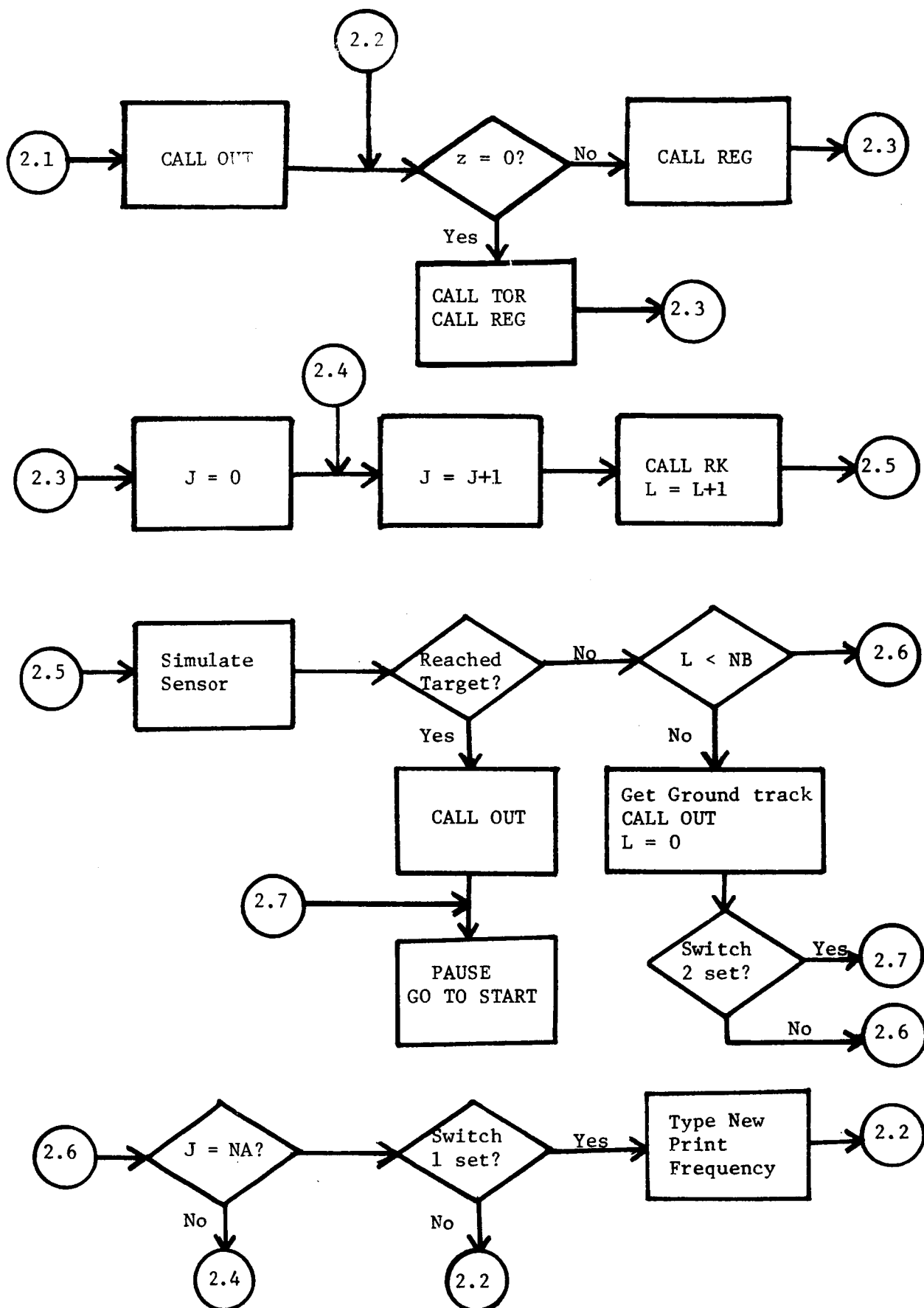


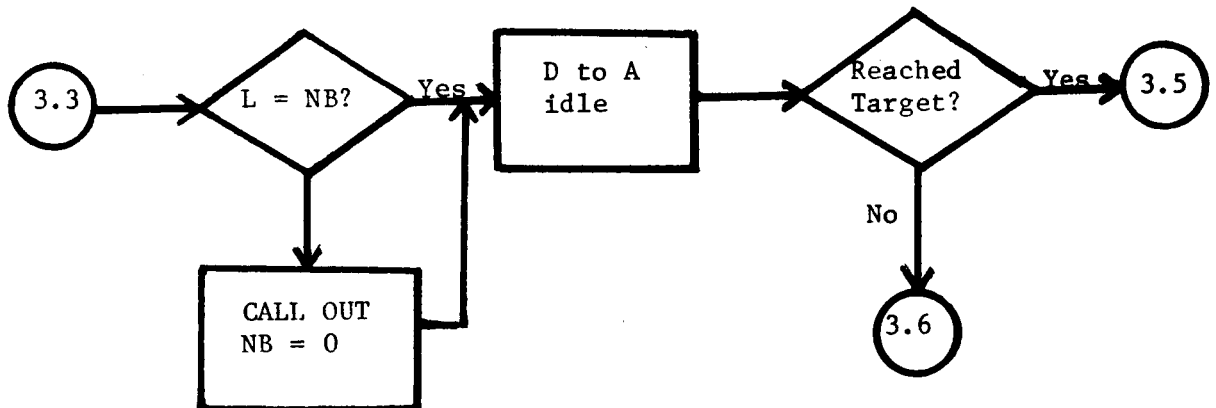
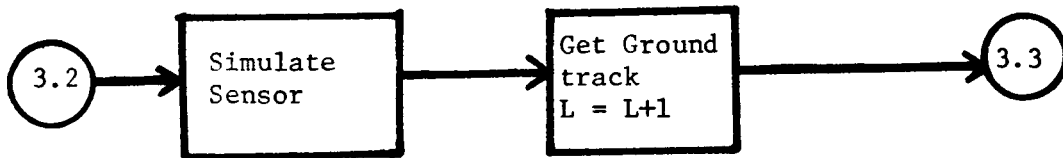
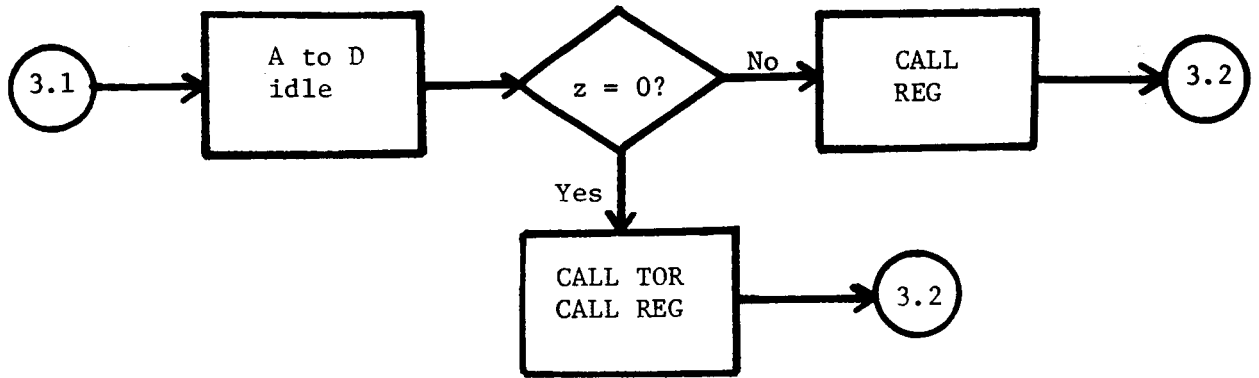
Figure 7.2.4.1

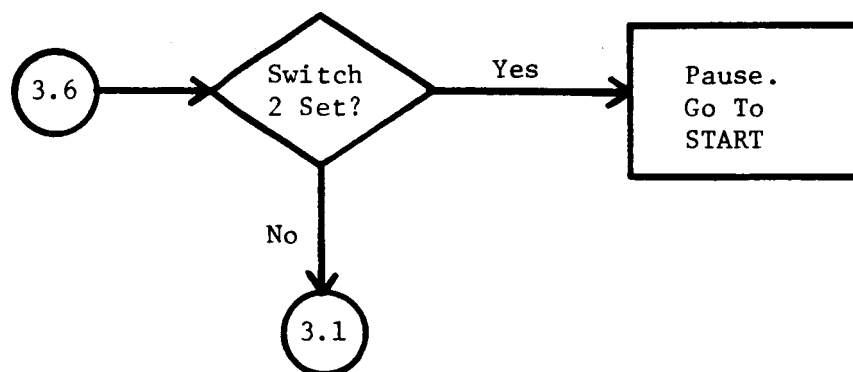
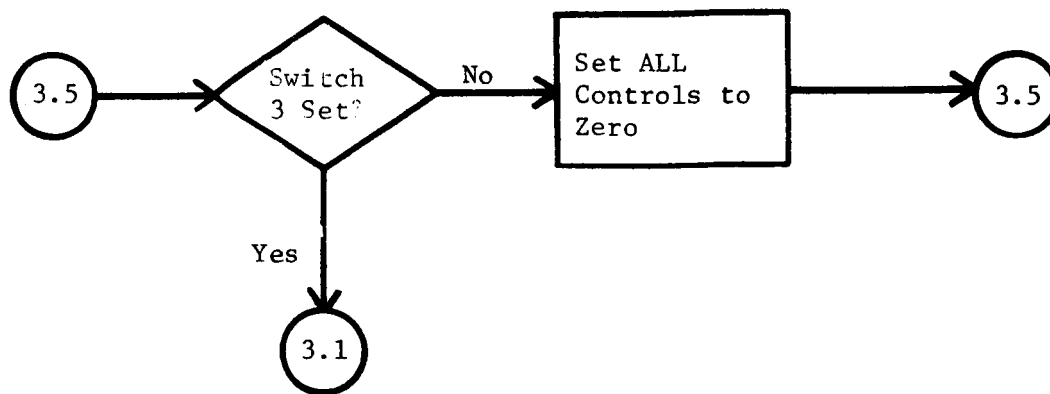
### 7.3 *Flow Charts*

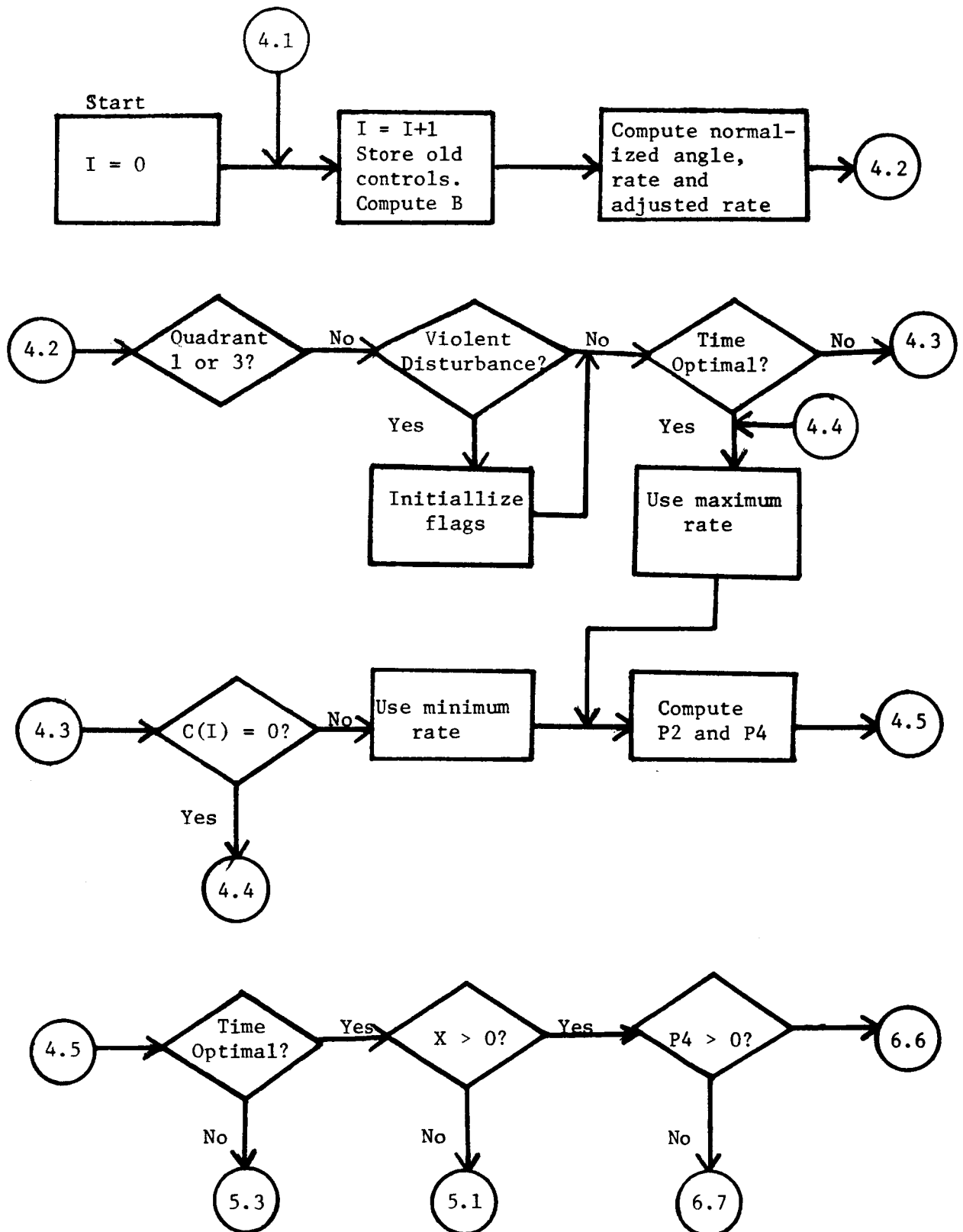
The flow charts which follow are for the main program and subroutine "REG" only. All other subroutines are straight-forward sequential computation and warrant no flow chart. The charts exhibited are those of the sensor model. The logic for the ideal model is identical, except that any reference to sensor evaluation should be omitted.

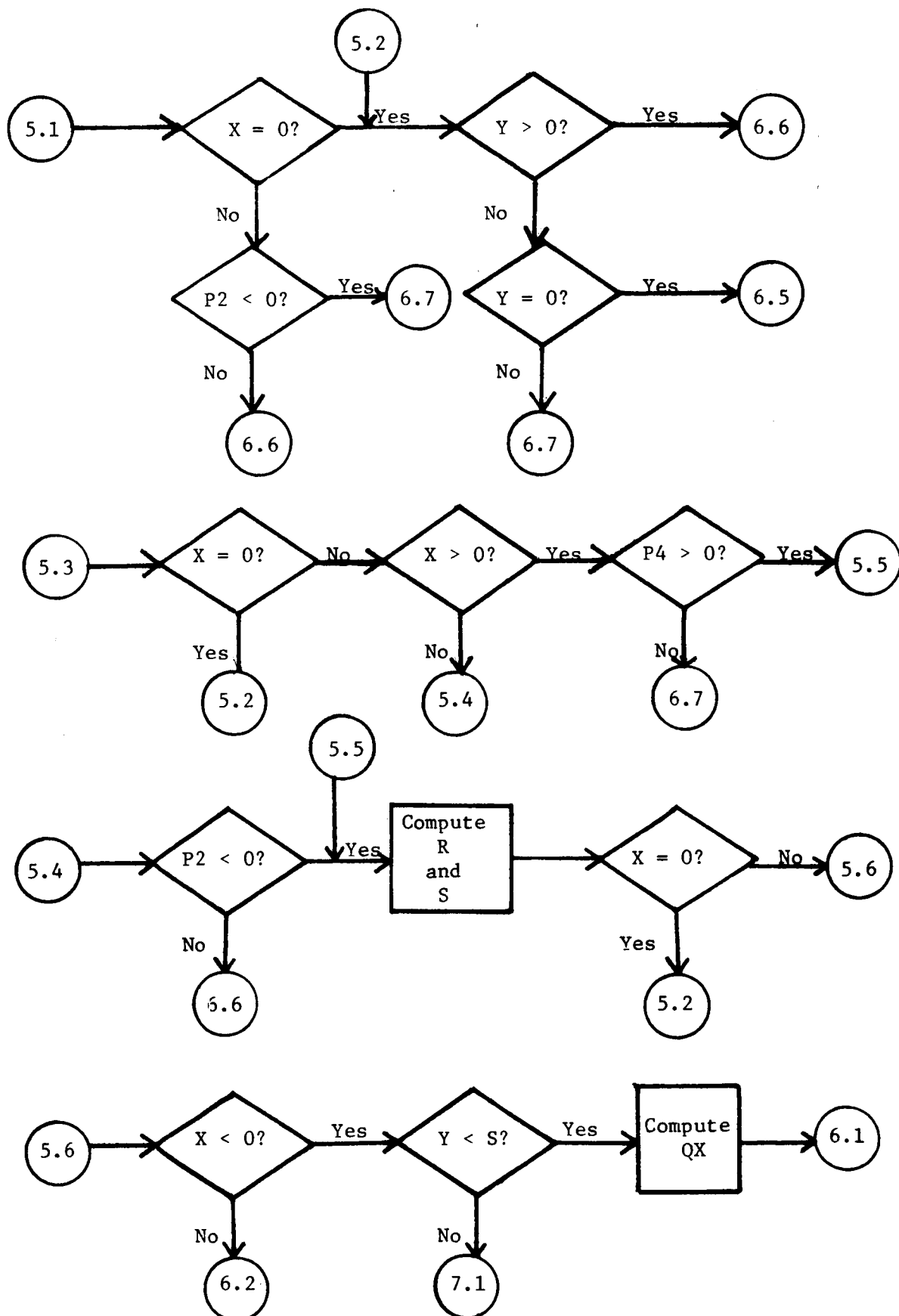
Main Program



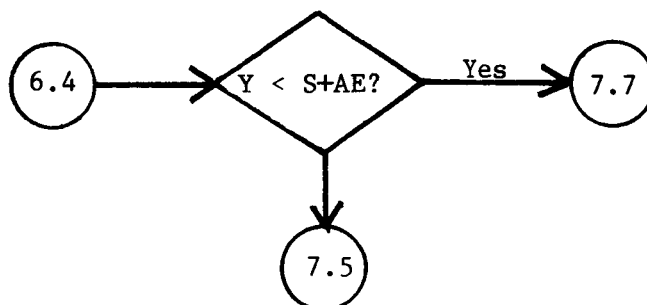
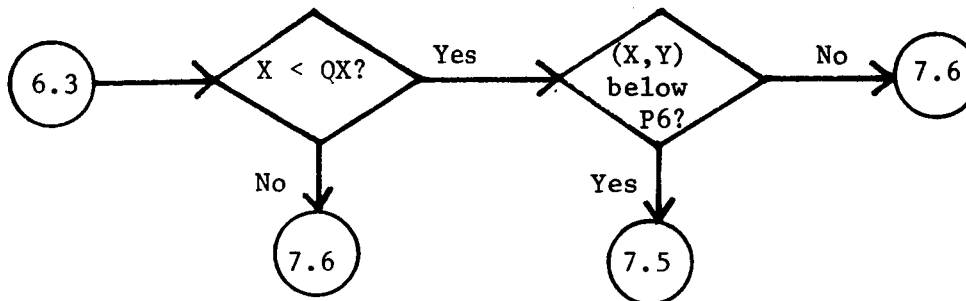
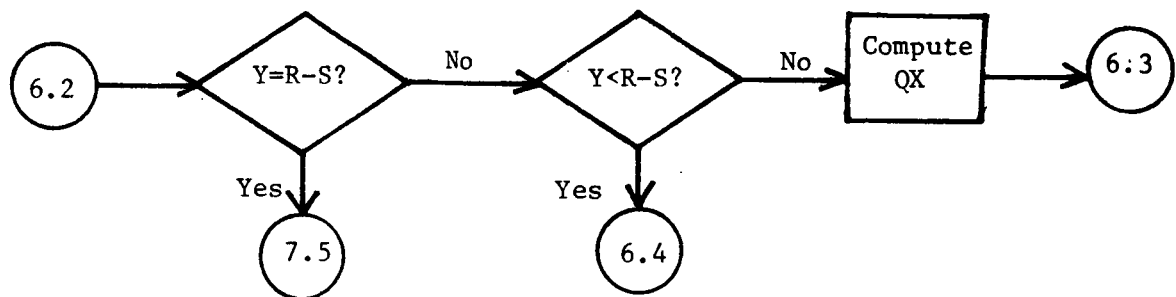
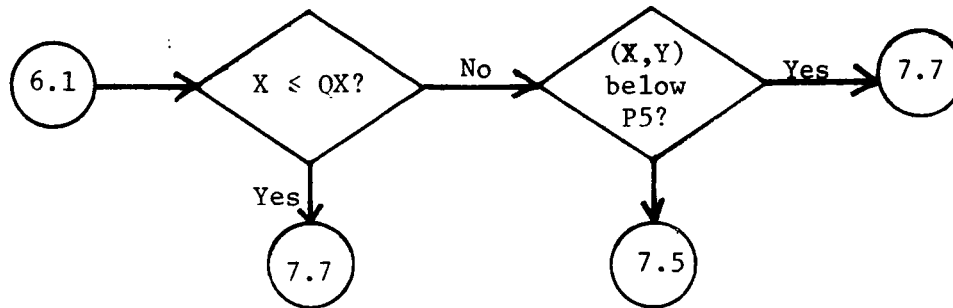


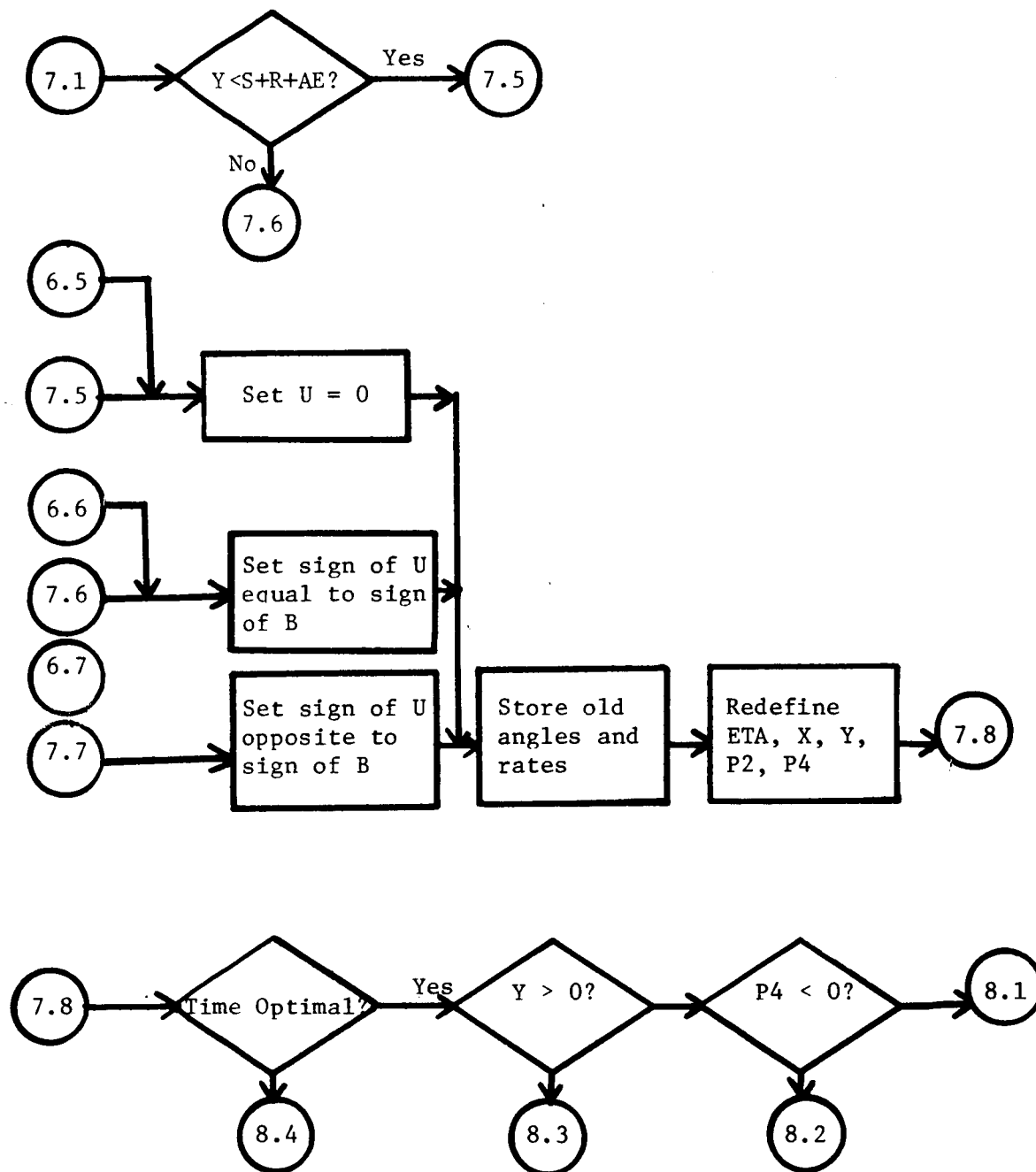


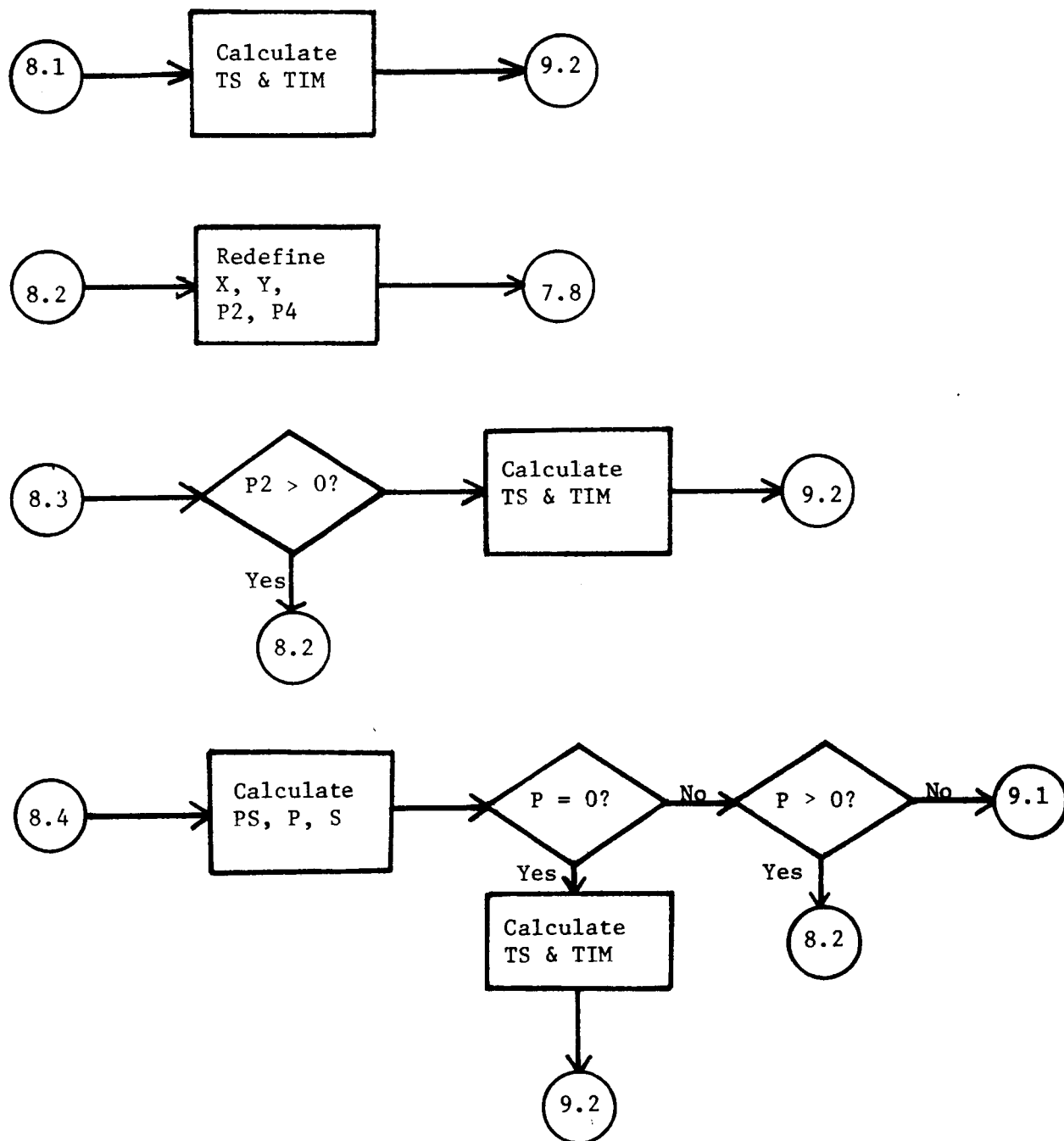
REG

REG

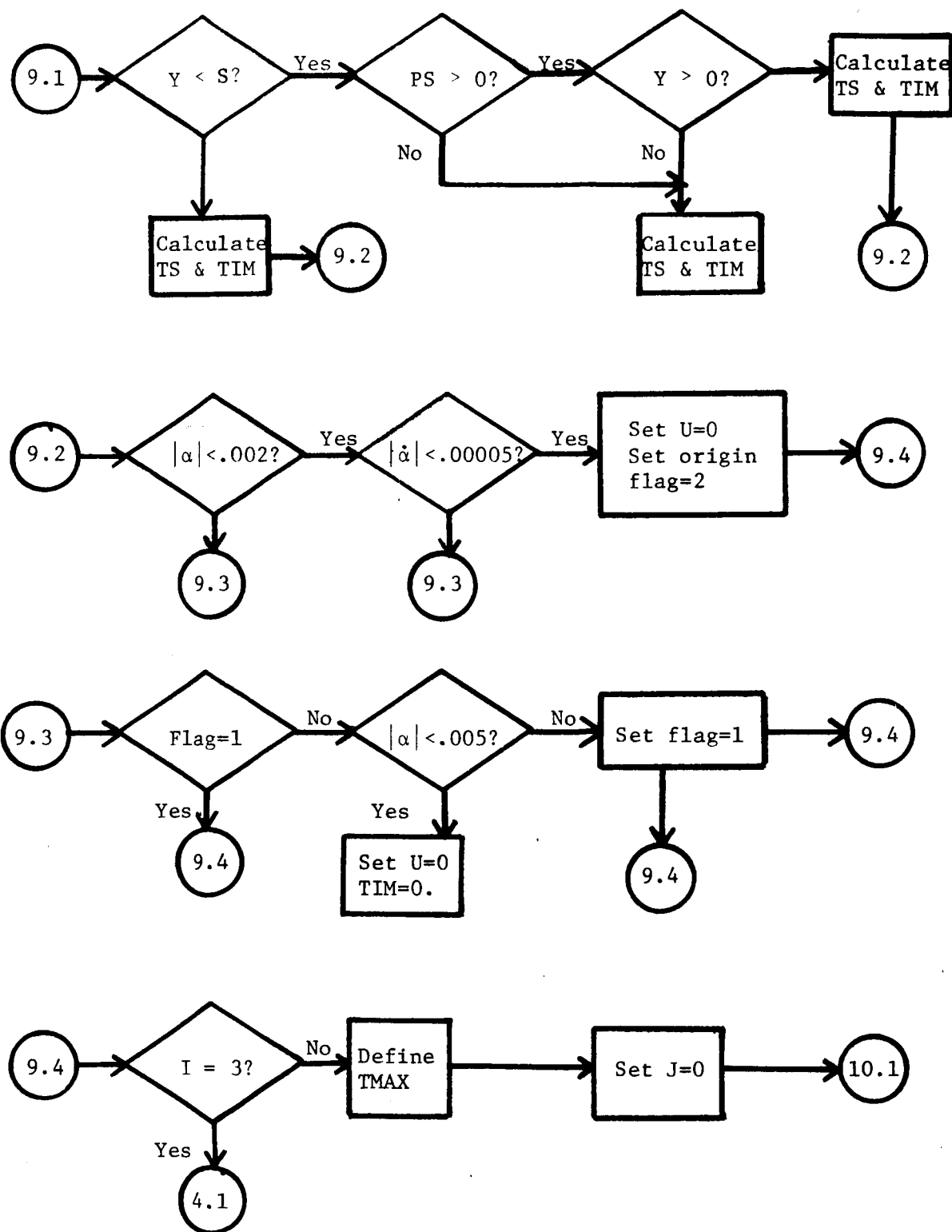


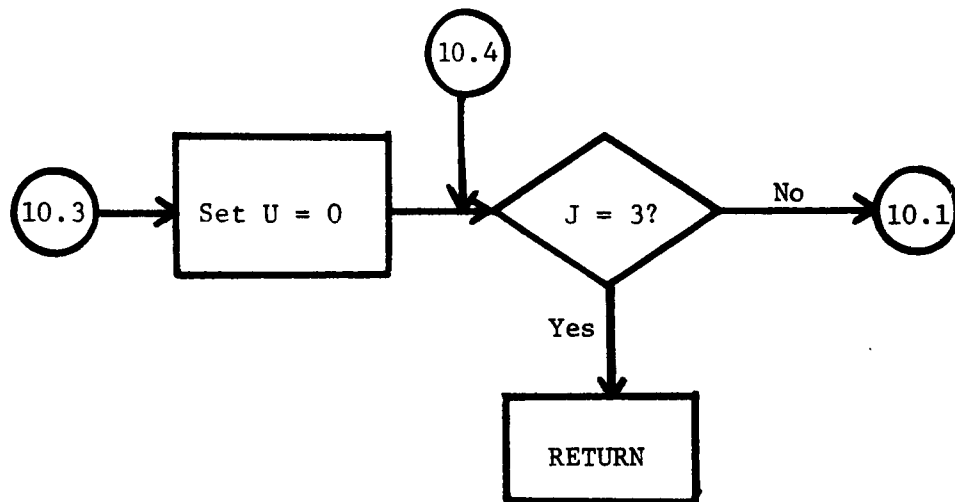
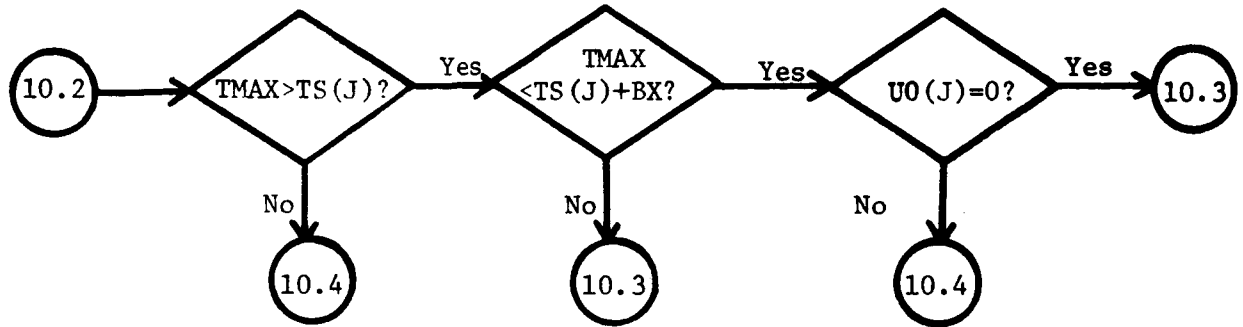
REG

REG

REG

REG

 $|\alpha|$  = angle $|\dot{\alpha}|$  = rate



#### 7.4 *Program Listings*

Three distinct computer programs appear on the following pages. The first is the ideal model program. The second program includes several real effects, i.e. those due to the sensors, solar torques and gravity gradient. Subroutine "TOR" in this listing is the subroutine which evaluates the later two quantities, whereas in the ideal program "TOR" is a dummy subroutine. The third program listing is one for simulating elastic effects on the satellite. Because of storage limitations this is simply a digital program, no hybrid operation being possible, nor does it take into consideration any other real effects.

C  
C  
C  
C  
C

# IDEAL MODEL

DATE 3 MAR 68

```

DIMENSION ELV(3,2),AZ(3,2)
DIMENSION X(6),U(3),T(3),ZI(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(6)
DIMENSION RA(3,3),RB(3,3),RC(3,3),RD(3,3),ETA(3),C(3),NF(3)
DIMENSION AP(3),XX(6)
COMMON X,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI/HYB/PA,PB,PC,AC
COMMON /CON/C,ETA,NF,XF,AE,AX,BX
COMMON /MISAL/ TH(3,3,2),UX(3)
COMMON /POSIN/XLAT,XLON,PIR,ROR,PBS,YBS,RBS,PIB,YB,R0B
COMMON /TEMP/T1,T2,T3,T4,T5,T6,T7,T8,T9,T0
13 TI=0.; L=0; T4=6.610808; NC=2; RAD=.017453      ;AC=1.
PRINT 113
DO 1 I=1,3
NF(I)=1.; U(I)=ZM(I); C(I)=1.
1 CONTINUE
READ 100,NA,NB,PA,PB,PC,Z
READ 101,(ZI(I),I=1,3),(ZM(I),I=1,3),(AL(I),I=1,3)
READ 101,((ZL(I,J),J=1,3),I=1,3)
READ 101,AX,BX,XF,PIB,R0B,YB,RDBS,YDBS,PDBS
READ 101,Y0,AT,ONG,XLAT,XLON,DT,AE
IF(Z.NE.0.) READ 101,T(1),T(2),T(3)
C FLV(I,J)=ELEVATION IN DEGREES FOR AXIS I,J=1 PLUS,2 MINUS
C AZ(I,J)=AZIMUTH
READ 101,(ELV(I,1),I=1,3),(AZ(I,1),I=1,3)
READ 101,(ELV(I,2),I=1,3),(AZ(I,2),I=1,3)
DO 40 J=1,2
DO 40 I=1,3
FLV(I,J)=ELV(I,J)*RAD
40 AZ(I,J)=AZ(I,J)*RAD
DO 42 K=1,2
DO 41 I=1,3
41 TH(1,1,K)=COS(ELV(1,K))*COS(AZ(1,K))
TH(2,1,K)=SIN(ELV(1,K))
TH(3,1,K)=COS(ELV(1,K))*SIN(AZ(1,K))
TH(1,2,K)=COS(ELV(2,K))*SIN(AZ(2,K))
TH(3,2,K)=SIN(ELV(2,K))
TH(1,3,K)=SIN(ELV(3,K))
42 TH(2,3,K)=COS(ELV(3,K))*SIN(AZ(3,K))
PRINT 120
PRINT 101,(ELV(I,1)/RAD,I=1,3),(AZ(I,1)/RAD,I=1,3)
PRINT 101,(ELV(I,2)/RAD,I=1,3),(AZ(I,2)/RAD,I=1,3)
PRINT 117
PRINT 118,((TH(I,J,1),J=1,3),I=1,3)
PRINT 119
PRINT 118,((TH(I,J,2),J=1,3),I=1,3)
PRINT 113
117 FORMAT (1H1,5X, $MULTIPLIERS FOR POSITIVE THRUSTS,///)
118 FORMAT (3F15.8)
119 FORMAT (1H0,5X, $MULTIPLIERS FOR NEGATIVE THRUSTS,///)

```

```

120 FORMAT (1H1, SELEVATION 1-2-3: AZIMUTH 1-2-3: POSITIVE AXES LINE 1
1. NEGATIVE AXES LINE 2.$ )
ZIM(1)=ZI(1)/ZM(1); ZIM(2)=ZI(2)/ZM(2); ZIM(3)=ZI(3)/ZM(3)
IF(ZL(1,2).EQ.0..AND.ZL(1,3).EQ.0..) NC=1
IF (SENSE SWITCH 4) 2,23
2 WRITE(102,102); READ(101,103)YO,AT,ONG,XLAT,XLON
23 T7=AT; T8=ONG; T9=XLAT; T10=XLON; ONG=ONG-AL(3); XLON=XLON-AL(3)
3 AT=AT*RAD; ONG=ONG*RAD; YO=YO*RAD; SAT=SIN(AT); SON=SIN(ONG)
CAT=COS(AT); CON=COS(ONG)
T1=SIN(YO); T2=COS(YO); T3=T4-CAT*CON
P2=ATAN(-CAT*SON/T3); SP2=SIN(P2)
R2=ATAN(SAT/(COS(P2)*T3-CAT*SON*SP2) )
T5=T1*COS(R2)
YBS=ATAN(T5/SQRT(1.-T5**2))
T5=SIN(R2)/COS(YBS)
RBS=ATAN(T5/SQRT(1.-T5**2))
T5=(SP2*T2-T1*SIN(R2)*COS(P2))/COS(YBS)
PBS=ATAN(T5/SQRT(1.-T5**2))
PRINT 105; PRINT 106,((ZL(I,J), J=1,3), I=1,3)
PRINT 107; PRINT 106,YO/RAD,T7,T8,T9,T10,RBS,YBS,PBS,DT
PRINT 108; PRINT 106,(ZI(I), I=1,3),(ZM(I), I=1,3),(AL(I), I=1,3)
PRINT 109; PRINT 106,AX,BX,XF,AE; CALL INIT
CALL STUP(0.,.35692,0.,RA); CALL STUP(AL(3),0.,0.,RB)
DO 4 I=1,3; DO 4 J=1,3; RD(I,J)=0.; DO 4 K=1,3
4 RD(I,J)=RD(I,J)+RA(I,K)*RB(K,J)
CALL STUP(PIR,ROR,0.,RA)
DO 5 I=1,3; DO 5 J=1,3; RB(J,I)=0.; DO 5 K=1,3
5 RB(J,I)=RB(J,I)+RD(I,K)*RA(K,J)
C 5 TAKES THE TRASPOSE AS IT COMPUTES THE PRODUCT
CALL STUP(PIR,ROR,YB,RA); SPIR=SIN(PIR); SROR=SIN(ROR)
CPIR=COS(PIR); CROR=COS(ROR)
IF(7.NE.0.) GO TO 16
6 CALL TOR (RA,RB,SPIR,SROR,CPIR,CROR)
16 CONTINUE
T4=T7
T5=T8
T7=T(1)*PA/ZI(1); T8=T(2)*PA/ZI(2); T9=T(3)*PA/ZI(3)
PRINT 110; PRINT 106,T(1),T(2),T(3),T7,T8,T9
IF (NA)15,14,15
15 PRINT 111; PRINT 115; PRINT 116; CALL OUT; GO TO 10
14 PRINT 112; PRINT 106,T1,X(1)*PB,X(2)*PB,X(3)*PB,X(4)*PC,
1X(5)*PC,X(6)*PC
PRINT 113; WRITE(102,114); PAUSE 11
100 FORMAT (2I5,5F10.5)
101 FORMAT (9F8.5)
102 FORMAT($ TYPE POS YO,AT,ONG,TARGET LAT AND LONG 9F5.3$ )
103 FORMAT (9F5.3)
105 FORMAT (1H0,2X 10HLAMDA(1,1),4X10HLAMDA(1,2),4X10HLAMDA(1,3),4X10H-
1LAMDA(2,1),4X10HLAMDA(2,2),4X10HLAMDA(2,3),4X10HLAMDA(3,1),4X10HLA-
2MDA(3,2),4X10HLAMDA(3,3) )
106 FORMAT(1X,1P9E14.5)
107 FORMAT (1H0,4X2HY0,12X2HAT,11X3HONG,11X3HLAT,11X3HLON,11X3HRBS,11X
23HYRS,11X3HPBS,12X2HDT )
108 FORMAT (1H0,4X8MMOMENT 1.6X8MMOMENT 2.6X8MMOMENT 3.8X4HZM 1,10X4HZ

```



```

      2M 2.10X4HZM 3.8X7HALPHA 1.7X7HALPHA 2.7X7HALPHA 3 )
109 FORMAT (1H0,5X2HAX,12X2HBX,12X2HXF,12X2HAE )
110 FORMAT (1H0,4X8HTORQUE 1.6X8HTORQUE 2.6X8HTORQUE 3.15X14HSCALED TO
      1RQUS )
111 FORMAT (1H0, 5X 4HTIME, 10X4HX(1), 10X4HX(2), 10X 4HX(3), 10X
      14HX(4), 10X 4HX(5), 10X 4HX(6) )
112 FORMAT (1H0,4X4HTIME, 19X 26HSCALED VALUES OF X(1-6) )
113 FORMAT (1H1 )
114 FORMAT (10HSET ANALOG )
115 FORMAT (20X 4HD(1), 10X 4HD(2), 10X 4HD(3), 10X 4HD(4),
      110X 4HD(5), 10X 4HD(6) )
116 FORMAT (20X 4HU(1), 10X 4HU(2), 10X 4HU(3), 10X 3HX1S, 11X 3HX4S,
      110X 1HS )
      7 CALL CONDITION
          CONNECT(40,ADCIN); CONNECT(43,DT0A); CONNECT(44,AT0D)
      CALL ENABLE; CALL ARM(40B,43B,44B); AC=1.; CALL REG; CALL DT0A
S      F0M 030010
      8 CONTINUE
S      F0M 030013
      CALL IDLE; AC=0.
24 IF(7.NE.0.) CALL REG; GO TO 19
      9 CALL TOR(RA,RB,SPIR,SROR,CPIR,CROR); CALL REG
19 CONTINUE
      ASSIGN 32 TO NG
      GO TO 30
32 CONTINUE
      I=L+1
      IF(L.GE.NB)CALL OUT; L=0
      AC=1.; CALL IDLE
      IF(NF(1).EQ.NF(2).EQ.NF(3).EQ.2) GO TO 21
C THIS IS THE DT0A IDLE
22 IF(SENSE SWITCH 2) 20,8
21 IF(SENSE SWITCH 3) 8,25
25 U(1)=0; U(2)=0; U(3)=0; CALL DT0A; GO TO 21
20 PAUSE 16; GO TO 13
10 IF(7.NE.0.) CALL REG; GO TO 17
      CALL TOR(RA,RB,SPIR,SROR,CPIR,CROR); CALL REG
17 CONTINUE
      DO 11 J=1,NA
      CALL RK; L=L+1
      IF(NF(1).EQ.NF(2).EQ.NF(3).EQ. 2) CALL OUT; GO TO 20
27 IF(L.LT.NB)GO TO 11
      ASSIGN 31 TO NG
      GO TO 30
31 CALL OUT
      I=0
      IF (SENSE SWITCH 2) 20,11
11 CONTINUE
      IF(SENSE SWITCH 1) 12,10
C SET SWITCH 1 FOR NEW PRINT FREQUENCY
12 READ(101,100)NB; GO TO 10
C FOLLOWING COMPUTES GROUND TRACK, WHERE T4 IS LAT AND T5 IS LONG
30 CONTINUE
      S1=SIN(X(3))

```

```

S2=SIN(X(2))
S3=SIN(X(1))
C1=COS(X(3))
C2=COS(X(2))
C3=COS(X(1))
S7=SROR*S1+C2+CROR*S2
C7=SQRT(ABS(1.-S7**2))
S8=(SROR*(C1+C3-S1*S2*S3)+CROR*S3*C2)/C7
C8=SQRT(ABS(1.-S8**2))
S6=(SPIR*C1*C2+CPIR*CROR*S1*C2-CPIR*SROR*S2)/C7
C6=SQRT(ABS(1.-S6**2))
F2=ATAN(S7/C7/C8)
T2=SIN(F2)
D2=COS(F2)
T1=S6*C7*D2+T2*(C6*S8+S6*C8*S7)
D1=SQRT(ABS(1.-T1**2))
T3=S8*C7
D3=SQRT(ABS(1.-T3**2))
Y1=T3*(ZK*D3*D1-SQRT(ABS(1.-ZK**2*(1.-D3**2*D1**2))))
Z1=SQRT(ABS(1.-Y1**2))
T4 =ATAN(Y1/Z1)/RAD
A1=SQRT(ABS(Z1**2-ZK**2*T1**2))
T5 =ATAN(T1*(A1-ZK*D1)/(A1+ZK*T1*T1)/D1)/RAD +AL(3)
GO TO NG.(32,31)
END

```

```

SUBROUTINE REG
C (1-17),60,(63-66),68,70,73,(101-129),134,135,138
C ,200-203),(207-217),(222-225),345
  DIMENSION AL(3),NF(3),ETA(3),U0(3),C(3),TIM(3),TS(3),L(6)
  DIMENSION XX(6),U(3),T(3),ZI(3),ZM(3),ZIM(3),X0(6),ZL(3,3),XA(6)
  COMMONXX,U,T,ZI,ZM,ZIM,AL,ZL,UT,NC,R,S,D,NA,TI
  COMMON /TEMP/T1,T2,T3,T4,T5,T6,T7,T8,T9,T0
  COMMON /C0N/C,ETA,NF,XF,AE,AX,9X
  COMMON /MISAL/ TH(3,3,2),UX(3)
  IF(U(1))92,91,91
01 K=1
  GO TO 93
02 K=2
03 IF(U(2))95,94,94
04 I=1
  GO TO 96
05 I=2
06 IF(U(3))98,97,97
07 M=1
  GO TO 99
08 M=2
09 UX(1)=TH(1,2,L)*U(2)+TH(1,3,M)*U(3)
  UX(2)=TH(2,1,K)*U(1)+TH(2,3,M)*U(3)
  UX(3)=TH(3,1,K)*U(1)+TH(3,2,L)*U(2)
  DO 115 I=1,3
200 U0(I) = U(I)
  GO TO (201,202,203),I
201 R=(T(1)+(ZI(2)-ZI(3))*XX(5)*XX(6)*AL(1))/ZM(1)
  R=R+UX(1)/ZM(1)
  T1=XX(4)-XX(2)*XX(6)*AL(1)
  GO TO 11
203 R= ( T(3)+(ZI(1)-ZI(2))*XX(4)*XX(5)*AL(1))/ZM(3)
  R=R+UX(3)/ZM(3)
  T1=XX(6)+XX(1)*XX(5)*AL(1)
11 X=XX(1)*ZIM(1)*SIGN(1.,-R)
  Y=XX(I+3)*ZIM(1)*SIGN(1.,-R)
  GO TO 18
202 R=(T(2)+(ZI(3)-ZI(1))*XX(6)*XX(4)*AL(1))/ZM(2)
  R=R+UX(2)/ZM(2)
  T1=XX(5)-XX(1)*XX(6)*AL(1)
  X=XX(1)*ZIM(1)*SIGN(1., R)
  IF (B.EQ.0.) X=-X
  Y=XX(I+3)*ZIM(1)*SIGN(1.,-R)
18 IF(X*Y)2,2,1
1 U(I)=SIGN(ZM(I),-X)*SIGN(1.,-B)
  FTA(I)=-1.
  C(I)=1.
  GO TO 10
2 T2=200.*D1
  IF(ZIM(1)*ABS(X0(I+3)-XX(I+3))-12)12,12,13
12 IF(ABS(X0(I)-XX(I))-12*ABS(XX(I+3)))14,14,13
13 FTA(I)=-1.
  C(I)=1.

```

```

      GO TO 9
14  C(I)=C(I)*U(I)
      9 CONTINUE
      8 GO TO(209,345),NC
C NC=1 TIME OPTIMAL.208 USES MIN.209 USES MAX.
345 IF(C(I))203,209,208
208 IF(ABS(XX(I+3))-ABS(T1))212,212,210
209 IF(ABS(XX(I+3))-ABS(T1))210,212,212
210 Y=SIGN(T1*ZIM(I),Y)
212 IF(ETA(I))207,207,213
213 Y=Y*XF
207 T3=Y*Y/2.
      KA = 1
      IF ( U(I).LE.0 ) KA = 2
      X1 = TH(I,1,KA)
      RA = -ABS(B)
      P2 = X - T3/(BA - X1 )
      P4 = X - T3/(BA + X1 )
      GO TO (63,60),NC
63 IF(X)64,66,65
64 IF(P2)17,16,16
65 IF(P4)17,17,16
66 IF(Y) 17,15,16
60 IF(X)68,66,70
68 IF(P2)73,16,16
70 IF(P4)17,17,73
73 CONTINUE
      T4=ABS(ZL(I,2)*B/ZL(I,3))
      R=SQRT(2.*T4)
      S=SQRT(ZL(I,1)/ZL(I,3)+T4)
      IF(X)3,66,5
      3 IF(Y-S) 19,4,4
19 IF(R.EQ.0.) QX=-10000.; GO TO 20
      RA=-ABS(B); QX=(BA*S*S-2.*R*S+R*R)/(2.*BA*(BA-X1 ))
20 IF(X-QX)17,17,21
21 G=-ZL(I,1)-ZL(I,3)*Y**2+ZL(I,2)*BA
      7=-2.*ZL(I,2)+ZL(I,3)*(Y**2-2.*(BA-X1)*X )
      IF((-BA*Z*Z+2.*G*Z +G*G)*Y*Y-2.*G*G*(BA-X1)*X)17,15,15
      4 IF(Y-S-R-AE)15,15,16
      5 IF(Y-(R-S)) 6,15,22
22 IF(R.EQ.0.) QX=10000.; GO TO 23
      RA=-ABS(B); QX=(BA*(R-S)**2-2.*R*S+R*R)/(2.*RA*(BA+X1))
23 IF(X-QX)24,16,16
24 G=-ZL(I,1)-ZL(I,3)*Y**2-ZL(I,2)*BA
      7=-2.*ZL(I,2)+ZL(I,3)*(Y**2-2.*(BA+X1)*X )
      IF((-BA*Z*Z+2.*G*Z +G*G)*Y*Y-2.*G*G*(BA+X1)*X)16,15,15
      6 IF(Y-(-S-AE))17,15,15
15 U(I)=0.
      GO TO 7
16 U(I)=SIGN(ZM(I),B)
      IF(R.EQ.0.) U(I)=-U(I)
      GO TO 7
17 U(I)=SIGN(ZM(I),-B)
      7 IF(1.EQ.2) ETA(I)=-U(I)*XX(I); GO TO 10

```

```

      FTA(I)=U(I)*XX(I)
      10 X0(I)=XX(I); X0(I+3)=XX(I+3)
C 123 REGINS TIME SYNCH
      123 X=XX(I)*ZIM(I)
          IF (I.EQ.2) X=-X
          Y=XX(I+3)*ZIM(I)
          T8=U(I)
      121 T5=Y*Y/2.
          P2=X-T5
          P4=X+T5
          GO TO(117,101),NC
      117 IF(Y)107,107,109
      107 IF(P2)110,110,111
      111 X=-X
          Y=-Y
          T8=-U(I)
          GO TO 121
      109 IF(P4)112,111,111
      110 TS(I)=-1.
          GO TO 113
      112 TS(I)=ABS(Y/2.)-X/Y
          IF(T8*Y.LE.0.) TS(I)=-1.
      113 TIM(I)=2.*SQRT(ABS(P2))-Y
          GO TO 211
      101 PS=X+(.5+2.*ZL(I,2)/(ZL(I,1)-ZL(I,3)*Y**2))*Y**2
          P=Y*ABS(Y)+2.*X
          S=SQRT(ZL(I,1)/ZL(I,3))
          IF(P)87,79,111
      87 IF(Y-S)74,89,89
      89 TS(I)=-1.
          TIM(I)=Y-P4/S
          GO TO 211
      74 IF(PS)75,75,76
      75 T9=Y**2-2.*X
          T8=2.*ZL(I,1)+4.*ZL(I,2)
          T0=SQRT((T8+ZL(I,3)*T9)**2-8.*ZL(I,1)*ZL(I,3)*T9)
          T0=SQRT((T8+ZL(I,3)*T9-T0)/4./ZL(I,3))
C TO IS OMEGA S1
          TIM(I)=T0-P2/T0-Y
C REGION I
          IF(Y)77,77,78
      77 TS(I)=-1.
          GO TO 211
      78 TS(I)=Y/2.-X/Y
          IF(T8*Y.LE.0.) TS(I)=-1.
          GO TO 211
      76 IF (Y) 75,75,79
      79 TIM(I)=          Y/2.-X/Y
          GO TO 77
      211 IF(ABS(XX(I))-.0020) 222,223,223
      222 IF(ABS(XX(I+3))-.00005) 224,223,223
      224 U(I)=0.
          NF(I)=2
          TIM(I)=0.

```

```

      GO TO 115
223 GO TO (115,216) , NF(I)
216 IF(ABS(XX(I)) - .005)227,225,225
227 IF(ABS(XX(I+3)) -.000125) 226,225,225
226 U(I)=0.
      TIM(I)=0.
      GO TO 115
225 NF(I)=1
115 CONTINUE
127 TMAX=AMAX(TIM(1),TIM(2),TIM(3))*AX
129 DO 124 J=1,3
128 GO TO (138,124),NF(J)
138 IF(TS(J))124,125,125
125 IF(TMAX-TS(J))124,124,134
134 IF(TMAX-TS(J)-BX)135,126,126
135 IF(U0(J))124,126,124
126 U(J)=0.
124 CONTINUE
      RETURN
      END

```

```

SUBROUTINE DT0A
DIMENSION X(6),U(3),T(3),ZI(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(6)
COMMON /TEMP/T1,T2,T3,T4,T5,T6,T7,T8,T9,T0
COMMON X,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI/HYB/PA,PB,PC,AC
1 IF(AC)1,10,2
2 CALL DAC (0,T(1)*PA/ZI(1))
  CALL DAC (1,T(2)*PA/ZI(2))
  CALL DAC (2,T(3)*PA/ZI(3))
  CALL DAC(3,.01*T4)
  CALL DAC(4,.006*T5)
  IF(U(1))3,4,5
3 CONTINUE
S   F0M 034110
S   P0T =034110
S   F0M 034110
S   P0T =034101
  GO TO 6
4 CONTINUE
S   F0M 034110
S   P0T =034110
S   F0M 034110
S   P0T =034111
  GO TO 6
5 CONTINUE
S   F0M 034110
S   P0T =034100
S   F0M 034110
S   P0T =034111
6 IF(U(2))7,8,9
7 CONTINUE
S   F0M 034110
S   P0T =034112
S   F0M 034110
S   P0T =034103
  GO TO 11
8 CONTINUE
S   F0M 034110
S   P0T =034112
S   F0M 034110
S   P0T =034113
  GO TO 11
9 CONTINUE
S   F0M 034110
S   P0T =034102
S   F0M 034110
S   P0T =034113
11 IF(U(3))12,13,14
12 CONTINUE
S   F0M 034110
S   P0T =034114
S   F0M 034110
S   P0T =034105
  GO TO 15

```

```

13 CONTINUE
S   FOM 03411
S   POT =034
S   FOM 034110
S   POT =034115
    GO TO 14
14 CONTINUE
S   FOM 034110
S   POT =034114
S   FOM 034110
S   POT =034115
15 RETURN
10 PRINT 100
100 FORMAT (14H TIMING IS OFF )
    RETURN
    END

```



```

SUBROUTINE AT0D
DIMENSION X(6),U(3),T(3),ZI(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(6)
COMMON X,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,T1/HYB/PA,PB,PC,AC
CALL ADC (0,VA)
CALL ADC (3,VD)
CALL ADC (1,VB)
CALL ADC (4,VE)
CALL ADC (2,VC)
CALL ADC (5,VF)
X(1)=VA/PB; X(2)=VB/PB; X(3)=VC/PB
X(4)=VD/PC; X(5)=VE/PC; X(6)=VF/PC ;RETURN
END

```

```

SUBROUTINE OUT
  DIMENSION X(6),U(3),T(3),ZI(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(6)
  COMMON X,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI
  COMMON /TEMP/ T1,T2,T3,T4,T5,T6,T7,T8,T9,T0
  IF(NA)1,2,1
1  WRITE(6,112)TI,X,14,15
  WRITE(6,113)D
  WRITE(6,113)U, T(1),X(1)*ZIM(1),X(4)*ZIM(1),S
  RETURN
2  WRITE (6,114) TI,X,U
  RETURN
112 FORMAT (1H0,1P9E14.5)
113 FORMAT (15X, 1P7E14.5)
114 FORMAT (1H0, F6.2, 1P9E12.3)
  END

```

```

SUBROUTINE STUP (CP,CR,Y,CA)
DIMENSION CA(3,3)
SINR=SIN(CR)
SINR=SIN(CR); SINP=SIN(CP); SINY=SIN(Y)
COSR=COS(CR); COSP=COS(P); COSY=COS(Y)
C WF COMPUTE THE MATRIX I+8
CA(1,1)=COSP*COSY; CA(1,2)=-COSP*SINR*SINY-SINP*COSR
CA(1,3)=-COSP*COSR*SINY+SINP*SINR; CA(2,1)=SINP*COSY
CA(2,2)=-SINP*SINR*SINY+COSP*COSR
CA(2,3)=-SINP*COSR*SINY-COSP*SINR; CA(3,1)=SINY
CA(3,2)=SINR*COSY; CA(3,3)=COSR*COSY
RETURN
END

```

```

SUBROUTINE INIT
DIMENSION X(6),U(3),T(3),ZI(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(6)
COMMON X,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI
COMMON /POSIN/XLAT,XLON,PIR,ROR,PBS,YBS,PBS,PIB,YB,R0B
X(4)=0.; X(5)=0.; X(6)=0.
XLON = XLON *0.017453
XLAT =XLAT *0.017453
SILAT = SIN(XLAT)
SILON = SIN(XLON)
COLAT = COS(XLAT)
COLON = COS(XLON)
CONST=6.610808
PITCH =-ATANF((+SILON * COLAT)/(CONST - COLON * COLAT))
COP1 = COS(PITCH)
SINP1 = SIN(PITCH)
ROLL = ATANF (SILAT/((CONST -COLON*COLAT)* COP1 -SILON*COLAT *
1(SINP1))
PIR= PITCH
ROR =ROLL
CRBS = COS(RBS)
GO TO 200
200 CONTINUE
SRBS = SIN(RBS)
CYBS = COS(YBS)
SYBS = SIN(YBS)
CPIB = COS(PIB)
CYB = COS(YB)
CR0B = COS(R0B)
SYB = SIN(YB)
SR0B = SIN(R0B)
SPIB = SIN(PIB)
SPIR = SIN(PIR)
SR0R = SIN(R0R)
CPIR = COS(PIR)
CR0R = COS(R0R)
SPRB = SIN(PIR - PBS)
CPRB = COS(PIR - PBS)
G1 = SR0R * CYBS * SPRB + CR0R * SYBS
78 FORMAT (F9.5)
G2 = SQRTF (1.-G1**2)
X(2) = ATANF (G1/G2)
G3=(-CR0R*CYBS*SPRB+SR0R*SYBS)/G2
G4=SQRTF(1.-G3*G3)
X(3)=ATANF(G3/G4)
G6 =(-SR0R * CRBS * CPRB - SR0R* SRBS *SYBS *SPRB + CR0R*SRBS *
1(CYBS)/G2
G7= SQRTF (1 - G6**2)
X(1)= ATANF (G6/G7)
RETURN
END

```

```

SUBROUTINE DER
  DIMENSION X(6),U(3),T(3),ZI(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(6)
  COMMON X,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI
  COMMON /MISAL/ TH(3,3,2),UX(3)
  IF(U(1))2,1,1
1  K=1
  GO TO 3
2  K=2
3  IF(U(2)) 5,4,4
4  I=1
  GO TO 6
5  I=2
6  IF(U(3)) 8,7,7
7  M=1
  GO TO 9
8  M=2
9  DO 10 I=1,3
10 UX(I)=TH(I,1,K)*U(1)+TH(I,2,L)*U(2)+TH(I,3,M)*U(3)
  P(1)=-AL(1)* X(2)*X(6) + X(4)
  P(2)= AL(1) * X(1) *X(6) - X(5)
  P(3)= AL(1)*X(1)*X(5) + X(6)
  P(4)= (AL(1) *(ZI(2)-ZI(3))* X(5)* X(6) +UX(1) + T(1))/ZI(1)
  P(5) = (AL(1) *(ZI(3) -ZI(1))*X(4)*X(6)+UX(2) +T(2))/ZI(2)
  P(6)= (AL(1)* (ZI(1)-ZI(2))* X(4) * X(5) +UX(3) + T(3))/ZI(3)
  RETURN
  END

```

```
SUBROUTINE TOR(RA, RB, SPIR, SROR, CPIR, CROR)
  DIMENSION X(6), U(3), T(3), ZI(3), ZM(3), ZIM(3), AL(3), ZL(3,3), D(6)
  DIMENSION RA(3,3), RB(3,3), RC(3,3), RD(3,3), G(3,3), XI(3,3)
  COMMON X, U, T, ZI, ZM, ZIM, AL, ZL, DT, NC, R, S, D, NA, TI
  T(1)=1.E-7
  T(2)=1.E-7
  T(3)=1.E-7
  RETURN
  END
```

```

SUBROUTINE RK
  DIMENSION X(6),U(3),T(3),ZI(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(6)
  DIMENSION R(6),SA(6),SB(6),SC(6),SD(6)
  COMMON X,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,Z,S,D,NA,TI
  DO 2 N=1,6
2  R(N)=X(N)
  CALL DER
  DO 3 N=1,6
3  SA(N)=D(N)*DT
  DO 4 N=1,6
4  X(N)=R(N)+SA(N)/2.
  TI=TI+DT/2.
  CALL DER
  DO 5 N=1,6
5  SB(N)=D(N)*DT
  DO 6 N=1,6
6  X(N)=R(N)+SB(N)/2.
  CALL DER
  DO 7 N=1,6
7  SC(N)=D(N)*DT
  TI=TI+DT/2.
  DO 8 N=1,6
8  X(N)=R(N)+SC(N)
  CALL DER
  DO 9 N=1,6
  SD(N)=D(N)*DT
9  X(N)=R(N)+(SA(N)+SD(N))/6.+(SB(N)+SC(N))/3.
  RETURN
  END

```

SENSOR MODEL

DATE 16 OCT 67

C1 NA,NB,PA,PB,PC 215,5F10.5  
C2 Z1(1-3), ZM(1-3), ZL(1-3), 9F8.5  
C3 ZL(1,1), (1,2),.....(3,3) 9F8.5  
C4 AX,RX,XF,ZK1,ZK2,SD,PIB,R0B,YB 9F8.5  
C5 Y0,AT,ONG,XLAT,XLON,DT,AE 9F8.5  
C SW1 TYPE NEW PRINT FREQUENCY 15  
C SW2 READ NEW DATA AND RESTART  
C SWITCH 3 IGNORES FUEL CALCULATION  
C SW4 TYPE Y0, AT, ONG, XLAT, XLON 9F5.3  
C XF PREVENTS JAGS COMING IN; AE INCREASES ZERO BAND; AX,BX LEAD FACTORS  
C (1-25) (30-37) (100-116)

DIMENSION X(6),U(3),I(3),Z1(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(6)  
DIMENSION PA(3,3),RB(3,3),RC(3,3),RD(3,3),ETA(3),C(3),NF(3)  
DIMENSION AP(3),XX(6)  
DIMENSION ELV(3,2),AZ(3,2)  
COMMON /MISAL/ TH(3,3,2),UX(3)  
COMMON X,U,I,Z1,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,T1/HYB/PA,PB,PC,AC  
COMMON /CON/C,ETA,NF,XF,AE,AX,BX,XX  
COMMON /POSIN/XLAT,XLON,PIR,R0R,PBS,YBS,RBS,PIB,YB,R0B  
COMMON /TEMP/T1,T2,T3,T4,T5,T6,T7,T8,T9,T0  
13 I1=0.; L=0; T4=6.610806; NC=2; RAD=.017453 ;AC=1.  
PRINT 113  
ZK=T4  
DO 1 I=1,3  
NF(I)=1.; U(I)=ZM(I); C(I)=1.  
1 CONTINUE  
READ 100,NA,NB,PA,PB,PC,Z  
READ 101,(Z1(I),I=1,3),(ZM(I),I=1,3),(AL(I),I=1,3)  
READ 101,((ZL(I,J),J=1,3),I=1,3)  
READ 101,AX,BX,XF,ZK1,ZK2,SD,PIB,R0B,YB  
RK=1.-ZK1  
READ 101,Y0,AT,ONG,XLAT,XLON,DT,AE  
SIG=DT  
IF(NA.NE.0) SIG=SIG\*FLOAT(NA)  
S2=SIG\*\*2/2.  
IF(7.NE.0.) READ 101,T(1),T(2),T(3)  
FLV(I,J)=ELEVATION IN DEGREES FOR AXIS I,J=1 PLUS,2 MINUS  
AZ(I,J)=AZIMUTH  
READ 101,(ELV(I,1),I=1,3),(AZ(I,1),I=1,3)  
READ 101,(ELV(I,2),I=1,3),(AZ(I,2),I=1,3)  
DO 40 J=1,2  
DO 40 I=1,3  
FLV(I,J)=ELV(I,J)\*RAD  
40 AZ(I,J)=AZ(I,J)\*RAD  
DO 42 K=1,2  
DO 41 I=1,3



```

41 TH(1,1,K)=COS(ELV(1,K))*COS(AZ(1,K))
   TH(2,1,K)=SIN(ELV(1,K))
   TH(3,1,K)=COS(ELV(1,K))*SIN(AZ(1,K))
   TH(1,2,K)=COS(ELV(2,K))*SIN(AZ(2,K))
   TH(3,2,K)=SIN(ELV(2,K))
   TH(1,3,K)=SIN(ELV(3,K))
42 TH(2,3,K)=COS(ELV(3,K))*SIN(AZ(3,K))
   PRINT 120
   PRINT 101,(ELV(1,1)/RAD,I=1,3),(AZ(1,1)/RAD,I=1,3)
   PRINT 101,(ELV(1,2)/RAD,I=1,3),(AZ(1,2)/RAD,I=1,3)
   PRINT 117
   PRINT 118,((TH(1,J,1),J=1,3),I=1,3)
   PRINT 119
   PRINT 118,((TH(1,J,2),J=1,3),I=1,3)
   PRINT 113
117 FORMAT (1H1,5X, $MULTIPLIERS FOR POSITIVE THRUST$,///)
118 FORMAT (3F15.8)
119 FORMAT (1H0,5X, $MULTIPLIERS FOR NEGATIVE THRUST$,///)
120 FORMAT (1H1, $ELEVATION 1-2-3: AZIMUTH 1-2-3: POSITIVE AXES LINE 1
   1. NEGATIVE AXES LINE 2.$ )
   ZIM(1)=Z1(1)/ZM(1);ZIM(2)=Z1(2)/ZM(2);ZIM(3)=Z1(3)/ZM(3)
   IF(ZL(1,2).EQ.0..AND.ZL(1,3).EQ.0.) NC=1
   IF (SENSE SWITCH 4) 2,23
   2 WRITE(102,102); READ(101,103)Y0,AT,ONG,XLAT,XLON
23 T7=AT; T8=ONG; T9=XLAT; T10=XLON; ONG=ONG-AL(3); XLON=XLON-AL(3)
   3 AT=AT*RAD; ONG=ONG*RAD; Y0=Y0*RAD; SAT=SIN(AT); SON=SIN(ONG)
   CAT=COS(AT); CON=COS(ONG)
   T1=SIN(Y0); T2=COS(Y0); T3=T4-CAT*CON
   P2=ATAN(-CAT*SON/T3); SP2=SIN(P2)
   R2=ATAN(SAT/(COS(P2)*T3-CAT*SON*SP2) )
   T5=T1*COS(R2)
   YBS=ATAN(T5/SQRT(1.-T5**2))
   T5=SIN(R2)/COS(YBS)
   RBS=ATAN(T5/SQRT(1.-T5**2))
   T5=(SP2*T2-T1*SIN(R2)*COS(P2))/COS(YBS)
   PBS=ATAN(T5/SQRT(1.-T5**2))
   PRINT 105; PRINT 106,((ZL(1,J), J=1,3),I=1,3)
   PRINT 107; PRINT 106,Y0/RAD,T7,T8,T9,T10,RBS,YBS,PBS,DT
   PRINT 108;PRINT 106,(Z1(I),I=1,3),(ZM(I),I=1,3),(AL(I),I=1,3)
   PRINT 109;PRINT 106,AX,DX,XF,AE; CALL INIT
   DO 37 I=1,3
   XX(I)=X(I)
37 XX(I+3)=0.
   CALL STUP(0...35692,0.,RA); CALL STUP(AL(3),0.,0.,RB)
   DO 4 I=1,3; DO 4 J=1,3; RD(I,J)=0.; DO 4 K=1,3
   4 RD(I,J)=RD(I,J)+RA(I,K)*RB(K,J)
   CALL STUP(PIR,ROR,0.,RA)
   DO 5 I=1,3; DO 5 J=1,3; RB(J,I)=0.; DO 5 K=1,3
   5 RB(J,I)=RB(J,I)+RD(I,K)*RA(K,J)
C 5 TAKES THE TRASPOSE AS IT COMPUTES THE PRODUCT
   CALL STUP (PIB,R0B,YB,RA); SPIR=SIN(PIR); SROR=SIN(ROR)
   CPIR=COS(PIR); CROR=COS(ROR)
   IF(7.NE.0.) GO TO 16
   6 CALL TOR (RA,RB,SPIR,SROR,CPIR,CROR)

```

```

16 CONTINUE
  T4=T7
  T5=T8
  T7=T(1)*PA/ZI(1); T8=T(2)*PA/ZI(2); T9=T(3)*PA/ZI(3)
  PRINT 110; PRINT 106,T(1),T(2),T(3),T7,T8,T9
  IF (NA)15,14,15
15 PRINT 111; PRINT 115; PRINT 116; CALL OUT; GO TO 10
14 PRINT 112; PRINT 106,T1,X(1)*PB,X(2)*PB,X(3)*PB,X(4)*PC,
  1X(5)*PC,X(6)*PC
  PRINT 113; WRITE(102,114); PAUSE 11
100 FORMAT (2I5,5F10.5)
101 FORMAT (9F8.5)
102 FORMAT($ TYPE POS YO,AT,ONG,TARGET LAT AND LONG 9F5.3$ )
103 FORMAT (9F5.3)
105 FORMAT (1H0,2X 10HLAMDA(1,1),4X10HLAMDA(1,2),4X10HLAMDA(1,3),4X10H
  11AMDA(2,1),4X10HLAMDA(2,2),4X10HLAMDA(2,3),4X10HLAMDA(3,1),4X10HLA
  2MDA(3,2),4X10HLAMDA(3,3) )
106 FORMAT(1X,1P9E14.5)
107 FORMAT (1H0,4X2HY0,12X2HAT,11X3H0NG,11X3HLAT,11X3HL0N,11X3HRBS,11X
  23HYBS,11X3HPBS,12X2HDT )
108 FORMAT (1H0,4X8HM0MENT 1,6X8HM0MENT 2,6X8HM0MENT 3,8X4HZM 1,10X4HZ
  2M 2,10X4HZM 3,8X7HALPHA 1,7X7HALPHA 2,7X7HALPHA 3 )
109 FORMAT (1H0,5X2HAX,12X2HBX,12X2HXF,12X2HAE )
110 FORMAT (1H0,4X8HT0RQUE 1,6X8HT0RQUE 2,6X8HT0RQUE 3,15X14HSCALED TO
  1RQUES )
111 FORMAT (1H0, 5X 4HTIME, 10X4HX(1), 10X4HX(2), 10X 4HX(3), 10X
  14HX(4), 10X 4HX(5), 10X 4HX(6) )
112 FORMAT (1H0,4X4HTIME, 19X 26HSCALED VALUES 0F X(1-6) )
113 FORMAT (1H1 )
114 FORMAT (10HSET ANAL0G )
115 FORMAT (20X 4HD(1), 10X 4HD(2), 10X 4HD(3), 10X 4HD(4),
  110X 4HD(5), 10X 4HD(6) )
116 FORMAT (20X 4HU(1), 10X 4HU(2), 10X 4HU(3), 10X 3HX1S, 11X 3HX4S,
  110X 1HS )
  7 CALL CONDITION
      CONNECT(40,ADCIN); CONNECT(43,DT0A); CONNECT(44,AT0D)
  CALL ENABLE; CALL ARM(40B,43B,44B); AC=1.; CALL REG; CALL DT0A
S   F0M 030010
  8 CONTINUE
S   F0M 030013
  CALL IDLE; AC=0.
24 IF(Z.NE.0.) CALL REG; GO TO 19
  0 CALL T0R(RA,RB,SP1R,SR0R,CPIR,CR0R); CALL REG
19 ASSIGN 34 TO NG
  GO TO 35
34 CONTINU E
  ASSIGN 32 TO NG
  GO TO 30
32 CONTINUE
  I=I+1
  IF(L.GE.NB)CALL OUT; L=0
  AC=1.; CALL IDLE
  IF(NF(1).EQ.NF(2).EQ.NF(3).EQ.2) GO TO 21
C THIS IS THE DT0A IDLE

```

```

22 IF(SENSE SWITCH 2) 20.8
21 IF(SENSE SWITCH 3) 8.25
25 U(1)=0; U(2)=0; U(3)=0; CALL DT0A; GO TO 21
20 PAUSE 16; GO TO 13
10 IF(7.NE.0.) CALL REG; GO TO 17
CALL TOR(RA,RB,SPIR,SR0R,CPIR,CR0R); CALL REG
17 CONTINUE
DO 11 J=1,NA
CALL RK; L=L+1
ASSIGN 33 TO NG
GO TO 35
33 CONTINUE
IF(NF(1).EQ.NF(2).EQ.NF(3).EQ. 2) CALL OUT; GO TO 20
27 IF(L.LT.NB)GO TO 11
ASSIGN 31 TO NG
GO TO 30
31 CALL OUT
I=0
WRITE(6,106)XX,AP
IF (SENSE SWITCH 2) 20 ,11
11 CONTINUE
IF(SENSE SWITCH 1) 12,10
C SFT SWITCH_1 FOR NEW PRINT FREQUENCY
12 READ(101,103)NB; GO TO 10
C
C FOLLOWING IS RANDOM NUMBER GENERATOR
35 IA=ABS(IA*KA)
T1=FLOAT(IA)/CA
IA=ABS(IA*KA)
T2=FLOAT(IA)/CA
T9=SQRT(-2.*ALOG(T1))
T8=6.28318531*ALOG(T2)
T0=T9*COS(T8)
C
C FOLLOWING IS SENSOR PROGRAM
C AP ARE ALPHA PRIMES (ANGLES PLUS NOISE)
C XX(1-3) ARE FILTERED ANGLES;XX(4-6) ARE ESTIMATED RATES
DO 36 JA=1,3
AP(JA)=X(JA)+SD*T0
T1=XX(JA)+SIG*XX(JA+3)+S2*U(JA)/Z1(JA)
XX(JA)=T1*OK+ZK1*AP(JA)
36 XX(JA+3)=XX(JA+3)+SIG*U(JA)/Z1(JA)+ZK2*(AP(JA)-T1)
GO TO NG,(33,34)
C
C FOLLOWING COMPUTES GROUND TRACK, WHERE T4 IS LAT AND T5 IS LONG
30 CONTINUE
S1=SIN(X(3))
S2=SIN(X(2))
S3=SIN(X(1))
C1=COS(X(3))
C2=COS(X(2))
C3=COS(X(1))
S7=SR0R*S1*C2+CR0R*S2
C7=SQRT(ABS(1.-S7**2))

```

```

S8=(SR0R*(C1*C3-S1*S2*S3)+CR0R*S3*C2)/C7
CA=SQRT(ABS(1.-S8**2))
S6=(SPIR*C1*C2+CP1R*CR0R*S1*C2-CP1R*SR0R*S2)/C7
C6=SQRT(ABS(1.-S6**2))
F2=ATAN(S7/C7/CA)
T2=SIN(F2)
D2=COS(F2)
T1=S6*C7*D2+T2*(C6*S8+S6*C8*S7)
D1=SQRT(ABS(1.-T1**2))
T3=S8*C7
U3=SQRT(ABS(1.-T3**2))
Y1=T3*(ZK*D3*D1-SQRT(ABS(1.-ZK**2*(1.-D3**2*D1**2))))
Z1=SQRT(ABS(1.-Y1**2))
T4 =ATAN(Y1/Z1)/RAD
A1=SQRT(ABS(Z1**2-ZK**2*T1**2))
T5 =ATAN(T1*(A1-ZK*D1)/(A1+ZK*T1*T1)/D1)/RAD +AL(3)
DATA IA,KA,CA/1.4099,8388638./
GO TO NG.(32,31)
END

```

```

      SUBROUTINE REG
C (1-17),60,(63-66),68,70,73,(101-129),134,135,138
C ,200-203),(207-217),(222-225),345
      DIMENSION AL(3),NF(3),ETA(3),US(3),C(3),TIM(3),TS(3),D(6)
      DIMENSION XX(6),U(3),T(3),ZI(3),ZM(3),ZIM(3),X0(6),7L(3,3),XA(6)
      COMMON /CON/C,ETA,NF,XF,AE,AX,BX,XX
      COMMON XA,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,11
      COMMON /TEMP/T1,T2,T3,T4,T5,T6,T7,T8,T9,T0
      COMMON /MISAL/ TH(3,3,2),UX(3)
      IF(U(1))92,91,91
01 K=1
      GO TO 93
02 K=2
03 IF(U(2))95,94,94
04 I=1
      GO TO 96
05 I=2
06 IF(U(3))98,97,97
07 M=1
      GO TO 99
08 M=2
09 UX(1)=TH(1,2,L)*U(2)+TH(1,3,M)*U(3)
      UX(2)=TH(2,1,K)*U(1)+TH(2,3,M)*U(3)
      UX(3)=TH(3,1,K)*U(1)+TH(3,2,L)*U(2)
      DO 115 I=1,3
200 US(I) = U(I)
      GO TO (201,202,203),I
201 R=(T(1)+(ZI(2)-ZI(3))*XX(5)*XX(6)*AL(1))/ZM(1)
      R=R+UX(1)/ZM(1)
      T1=XX(4)-XX(2)*XX(6)*AL(1)
      GO TO 11
203 R= ( T(3)+(ZI(1)-ZI(2))*XX(4)*XX(5)*AL(1))/ZM(3)
      R=R+UX(3)/ZM(3)
      T1=XX(6)+XX(1)*XX(5)*AL(1)
11 X=XX(1)*ZIM(1)*SIGN(1.,-B)
      Y=XX(I+3)*ZIM(1)*SIGN(1.,-B)
      GO TO 18
202 R=(T(2)+(ZI(3)-ZI(1))*XX(6)*XX(4)*AL(1))/ZM(2)
      R=R+UX(2)/ZM(2)
      T1=XX(5)-XX(1)*XX(6)*AL(1)
      X=XX(1)*ZIM(1)*SIGN(1., B)
      IF (B.EQ.0.) X=-X
      Y=XX(I+3)*ZIM(1)*SIGN(1.,-B)
18 IF(X*Y)2,2,1
1 U(I)=SIGN(ZM(I),-X)*SIGN(1.,-B)
      FTA(I)=-1.
      C(I)=1.
      GO TO 10
2 T2=200.*DT
      IF(ZIM(1)*ABS(X0(I+3)-XX(I+3))-T2)12,12,13
12 IF(ABS(X0(I)-XX(I))-T2*ABS(XX(I+3)))14,14,13
13 FTA(I)=-1.
      C(I)=1.

```

```

      GO TO 9
14 C(I)=C(I)*U(I)
9 CONTINUE
8 GO TO (209,345),NC
C NC=1 TIME OPTIMAL.208 USES MIN.209 USES MAX.
345 IF(C(I))208,209,208
208 IF(ABS(XX(I+3))-ABS(T1))212,212,210
209 IF(ABS(XX(I+3))-ABS(T1))210,212,212
210 Y=SIGN(T1*ZIM(I),Y)
212 IF(ETA(I))207,207,213
213 Y=Y*XF
207 T3=Y*Y/2.
      KA = 1
      IF ( U(I).LE.0 ) KA = 2
      X1 = TH(I,1,KA)
      RA = -ABS(B)
      P2 = X - T3/(BA - X1 )
      P4 = X - T3/(BA + X1 )
      GO TO (63,60),NC
63 IF(X)64,66,65
64 IF(P2)17,16,16
65 IF(P4)17,17,16
66 IF(Y) 17,15,16
60 IF(X)68,66,70
68 IF(P2)73,16,16
70 IF(P4)17,17,73
73 CONTINUE
      T4=ABS(ZL(I,2)*B/ZL(I,3))
      R=SQRT(2.*T4)
      S=SQRT(ZL(I,1)/ZL(I,3)+T4)
      IF(X)13,66,5
3 IF(Y-S) 19,14
19 IF(B.EQ.0.) QX=-10000.; GO TO 20
      RA=-ABS(B) QX=(BA*S*S-2.*R*S+R*R)/(2.*BA*(BA-X1 ))
20 IF(X-QX)17,17,21
21 G=-ZL(I,1)-ZL(I,3)*Y**2+ZL(I,2)*BA
      Z=-2.*ZL(I,2)+ZL(I,3)*(Y**2-2.*(BA-X1)*X )
      IF((-BA*Z*Z+2.*G*Z +G*G)*Y*Y-2.*G*G*(BA-X1)*X)17,15,15
4 IF(Y-S-R-AE)15,15,16
5 IF(Y-(R-S)) 6,15,22
22 IF(B.EQ.0.) QX=10000.; GO TO 23
      RA=-ABS(B) QX=(BA*(R-S)**2-2.*R*S+R*R)/(2.*BA*(BA+X1))
23 IF(X-QX)24,16,16
24 G=-ZL(I,1)-ZL(I,3)*Y**2-ZL(I,2)*BA
      Z=-2.*ZL(I,2)+ZL(I,3)*(Y**2-2.*(BA+X1)*X )
      IF((-BA*Z*Z+2.*G*Z +G*G)*Y*Y-2.*G*G*(BA+X1)*X)16,15,15
6 IF(Y-(-S-AE))17,15,15
15 U(I)=0.
      GO TO 7
16 U(I)=SIGN(ZM(I),B)
      IF(B.EQ.0.) U(I)=-U(I)
      GO TO 7
17 U(I)=SIGN(ZM(I),-B)
7 IF(I.EQ.2) ETA(I)=-U(I)*XX(I); GO TO 10

```

```

      FTA(I)=U(I)*XX(I)
      10 X0(I)=XX(I); X0(I+3)=XX(I+3)
C 123 REGINS TIME SYNCH
      123 X=XX(I)*ZIM(I)
      IF (I.EQ.2) X=-X
      Y=XX(I+3)*ZIM(I)
      TA=U(I)
      121 T5=Y*Y/2.
      P2=X-T5
      P4=X+T5
      GO TO(117,101),NC
      117 IF(Y)107,107,109
      107 IF(P2)110,110,111
      111 X=-X
      Y=-Y
      TA=-U(I)
      GO TO 121
      109 IF(P4)112,111,111
      110 TS(I)=-1.
      GO TO 113
      112 TS(I)=ABS(Y/2.)*X/Y
      IF(T8*Y.LE.0.) TS(I)=-1.
      113 TIM(I)=2.*SQRT(ABS(P2))-Y
      GO TO 211
      101 PS=X+(.5+2.*ZL(I,2)/(ZL(I,1)-ZL(I,3)*Y**2))*Y**2
      P=Y*ABS(Y)+2.*X
      S=SQRT(ZL(I,1)/ZL(I,3))
      IF(P)87,79,111
      87 IF(Y-S)74,89,89
      89 TS(I)=-1.
      TIM(I)=Y-P4/S
      GO TO 211
      74 IF(PS)75,75,76
      75 T9=Y**2-2.*X
      TA=2.*ZL(I,1)+4.*ZL(I,2)
      T0=SQRT((T8+ZL(I,3)*T9)**2-8.*ZL(I,1)*ZL(I,3)*T9)
      T0=SQRT((T8+ZL(I,3)*T9-T0)/4./ZL(I,3))
C TO IS OMEGA S1
      TIM(I)=T0-P2/T0-Y
C REGION I
      IF(Y)77,77,78
      77 TS(I)=-1.
      GO TO 211
      78 TS(I)=Y/2.*X/Y
      IF(T8*Y.LE.0.) TS(I)=-1.
      GO TO 211
      76 IF (Y) 75,75,79
      79 TIM(I)= Y/2.*X/Y
      GO TO 77
      211 IF(ABS(XX(I))-.0020) 222,223,223
      222 IF(ABS(XX(I+3))-.00005) 224,223,223
      224 U(I)=0.
      NF(I)=2
      TIM(I)=0.

```

```

      GO TO 115
223 GO TO (115,216) , NF(I)
216 IF(ABS(XX(I))-.005)226,225,225
226 U(I)=0.
      TIM(I)=0.
      GO TO 115
225 NF(I)=1
115 CONTINUE
127 TMAX=AMAX(TIM(1),TIM(2),TIM(3))*AX
129 DO 124 J=1,3
128 GO TO (138,124),NF(J)
138 IF(TS(J))124,125,125
125 IF(TMAX-TS(J))124,124,134
134 IF(TMAX-TS(J)-BX)135,126,126
135 IF(U0(J))124,126,124
126 U(J)=0.
124 CONTINUE
      RETURN
      END

```



```

SUBROUTINE DT0A
  DIMENSION X(6),U(3),T(3),ZI(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(6)
  COMMON /TEMP/T1,T2,T3,T4,T5,T6,T7,T8,T9,T0
  COMMON X,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI/HYB/PA,PB,PC,AC
  1 IF(AC)1,10,2
  2 CALL DAC (0,T(1)*PA/ZI(1))
  CALL DAC (1,T(2)*PA/ZI(2))
  CALL DAC (2,T(3)*PA/ZI(3))
  CALL DAC(3,.01*T4)
  CALL DAC(4,.006*T5)
  IF(U(1))3,4,5
  3 CONTINUE
S   F0M 034110
S   P0T =034110
S   F0M 034110
S   P0T =034101
  GO TO 6
  4 CONTINUE
S   F0M 034110
S   P0T =034110
S   F0M 034110
S   P0T =034111
  GO TO 6
  5 CONTINUE
S   F0M 034110
S   P0T =034100
S   F0M 034110
S   P0T =034111
  6 IF(U(2))7,8,9
  7 CONTINUE
S   F0M 034110
S   P0T =034112
S   F0M 034110
S   P0T =034103
  GO TO 11
  8 CONTINUE
S   F0M 034110
S   P0T =034112
S   F0M 034110
S   P0T =034113
  GO TO 11
  9 CONTINUE
S   F0M 034110
S   P0T =034102
S   F0M 034110
S   P0T =034113
  11 IF(U(3))12,13,14
  12 CONTINUE
S   F0M 034110
S   P0T =034114
S   F0M 034110
S   P0T =034105
  GO TO 15

```

```

13 CONTINUE
S    FOM 034110
S    POT =034114
S    FOM 034110
S    POT =034115
      GO TO 15
14 CONTINUE
S    FOM 034110
S    POT =034104
S    FOM 034110
S    POT =034115
15 RETURN
10 PRINT 100
100 FORMAT (14H TIMING IS OFF )
      RETURN
      END

```

```

SUBROUTINE AT0D
DIMENSION X(6),U(3),T(3),Z1(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(6)
COMMON X,U,T,Z1,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI/HYB/PA,PB,PC,AC
CALL ADC (0,VA)
CALL ADC (3,VD)
CALL ADC (1,VB)
CALL ADC (4,VE)
CALL ADC (2,VC)
CALL ADC (5,VF)
X(1)=VA/PB; X(2)=VB/PB; X(3)=VC/PB
X(4)=VD/PC; X(5)=VE/PC; X(6)=VF/PC ;RETURN
END

```

```

SUBROUTINE OUT
  DIMENSION X(6),U(3),T(3),ZI(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(6)
  COMMON X,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI
  COMMON /TEMP/ T1,T2,T3,T4,T5,T6,T7,T8,T9,T0
  IF(NA)1,2,1
1 WRITE(6,112)T1,X,T4,T5
  WRITE(6,113)D
  WRITE(6,113)U, T(1),X(1)*ZIM(1),X(4)*ZIM(1),S
  RETURN
2 WRITE (6,114) T1,X,U
  RETURN
112 FORMAT (1H0,1P9E14.5)
113 FORMAT (15X, 1P7E14.5)
114 FORMAT (1H0, F6.2, 1P9E12.3)
  END

```

```

SUBROUTINE STUP (CP,CR,CY,CA)
DIMENSION CA(3,3)
SINR=SIN(CR)
SINR=SIN(CR); SINP=SIN(CP); SINY=SIN(Y)
COSR=COS(CR); COSP=COS(P); COSY=COS(Y)
C WF COMPUTE THE MATRIX I+B
CA(1,1)=COSP*COSY; CA(1,2)=-COSP*SINR*SINY-SINP*COSR
CA(1,3)=-COSP*COSR*SINY+SINP*SINR; CA(2,1)=SINP*COSY
CA(2,2)=-SINP*SINR*SINY+COSP*COSR
CA(2,3)=-SINP*COSR*SINY-COSP*SINR; CA(3,1)=SINY
CA(3,2)=SINR*COSY; CA(3,3)=COSR*COSY
RETURN
END

```

```

SUBROUTINE      INIT
DIMENSION X(6),U(3),T(3),Z1(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(6)
COMMON X,U,T,Z1,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI
COMMON /POSIN/XLAT,XLON,PIR,ROR,PBS,YBS,RBS,PIF,YB,ROB
X(3)=0.
X(4)=0.
X(5)=0.
XLON = XLON *0.017453
XLAT =XLAT *0.017453
SILAT = SINP(XLAT)
SILON = SINP (XLON)
COLAT = COSP (XLAT)
COLON = COSP (XLON)
CONST=6.610808
PITCH =-ATANP((+SILON * COLAT)/(CONST - COLON * COLAT))
COP1 = COSP (PITCH)
SINP1 = SINP (PITCH)
ROLL = ATANP (SILAT/((CONST -COLON*COLAT)* COP1 -SILON*COLAT *
1(SINP1))
PIR= PITCH
ROR =ROLL
CRBS = COSP(RBS)
GO TO 200
200 CONTINUE
SRBS = SINP (RBS)
CYBS = COSP (YBS)
SYBS = SINP (YBS)
CPIR = COSP (PIB)
CYR = COSP (YB)
CROR = COSP (ROB)
SYR = SINP (YB)
SROR = SINP (ROB)
SPIR = SINP (PIB)
SPIR = SINP (PIR)
SROR = SINP (ROR)
CPIR = COSP (PIR)
CROR = COSP (ROR)
SPRB = SINP (PIR - PBS)
CPRB = COSP (PIR - PBS)
G1 = SROR * CYBS * SPRB + CROR * SYBS
7A FORMAT (F9.5)
G2 = SQRTF (1.-G1**2)
X(2) = ATANP (G1/G2)
G3=(-CROR*CYBS*SPRB+SROR*SYBS)/G2
G4=SQRTF(1.-G3*G3)
X(3)=ATANP(G3/G4)
G6 =(-SROR * CRBS * CPRB - SROR* SRBS *SYBS *SPRB + CROR*SRBS *
1(CYBS)/G2
G7= SQRTF (1 - G6**2)
X(1)= ATANP (G6/G7)
RETURN
END

```

```

SUBROUTINE DER
  DIMENSION X(6),U(3),T(3),ZI(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(6)
  COMMON X,U,T,ZI,ZM,ZIM,AL,ZL,DI,NC,R,S,D,NA,I
  COMMON /MISAL/ TH(3,3,2),UX(3)
  IF(U(1))2,1,1
1  K=1
  GO TO 3
2  K=2
3  IF(U(2)) 5,4,4
4  I=1
  GO TO 6
5  I=2
6  IF(U(3)) 8,7,7
7  M=1
  GO TO 9
8  M=2
9  DO 10 I=1,3
10 UX(I)=TH(I,1,K)*U(1)+TH(I,2,L)*U(2)+TH(I,3,M)*U(3)
  D(1)=-AL(1)* X(2)*X(6) + X(4)
  D(2)= AL(1) * X(1) *X(6) - X(5)
  D(3)= AL(1)*X(1)*X(5) + X(6)
  D(4)= (AL(1) *(ZI(2)-ZI(3))* X(5)* X(6) +UX(1) + I(1))/ZI(1)
  D(5) = (AL(1) *(ZI(3) -ZI(1))*X(4)*X(6)+UX(2) +I(2))/ZI(2)
  D(6)= (AL(1)* (ZI(1)-ZI(2))* X(4) * X(5) +UX(3) + I(3))/ZI(3)
  RETURN
  END

```

```

SUBROUTINE TOR(RA,RB,SPIR,SROR,CPIR,CROR)
  DIMENSION X(6),UF(3),T(3),Z1(3)
  DIMENSION RA(3,3),RB(3,3),RC(3,3),PD(3,3),G(3,3),X1(3,3)
  COMMON X,UF,T,Z1
  FN=20.
  FL=13./4.
  GM=EL/225.
  FG2=4.*GM*GM
  TGL=2.*GM*EL
  S=0.4E-8
  SP=SINF(X(3))
  SY=SINF(X(2))
  SR=SINF(X(1))
  CP=COSF(X(3))
  CY=COSF(X(2))
  CR=COSF(X(1))
  SYBS=SROR*SP*CY+CROR*SY
  Z=SYBS*SYBS
31 CYBS=SQRTF(1.-Z)
  SRBS=(SROR*(CP*CR-SP*SR*SY)+CROR*SR*CY)/CYBS
  SPBS=(SPIR*CP*CY+CPIR*CROR*SP*CY-CPIR*SROR*SY)/CYBS
  Z=SRBS*SRBS
  YBS=ATANF(SYBS/CYBS)
  CRBS=SQRTF(1.-SRBS*SRBS)
  CPRS=SQRTF(1.-SPBS*SPBS)
  RBS=ATANF(SRBS/CRBS)
  PBS=ATANF(SPBS/CPRS)
  CALL STUP(PBS,PBS,YBS,RC)
C WF NEED RC TRANSPOSE.
  DO 51 NI=1,2
    IP=NI+1
    DO 51 NJ=IP,3
      TM=RC(NI,NJ)
      RC(NI,NJ)=RC(NJ,NI)
    GO TO 51
51 RC(NJ,NI)=TM
  DO 52 NI=1,3
    DO 52 NJ=1,3
      RD(NI,NJ)=0.
    DO 52 NK=1,3
52 RD(NI,NJ)=RD(NI,NJ)+RA(NI,NK)*RC(NK,NJ)
  DO 53 NI=1,3
    DO 53 NJ=1,3
      G(NI,NJ)=0.
    DO 53 NK=1,3
53 G(NI,NJ)=G(NI,NJ)+RD(NI,NK)*RB(NK,NJ)
C THIS COMPLETES THE G MATRIX.
  IF(G(1,2))1,99,1
C WF ASSUME ATAN(X) GIVES RANGE -90 TO +90 DEGREES
  DO 60 C09=0.
  IF(G(3,2))60,62,61
61 S19=1.
  GO TO 20

```



```

60 SIR=-1.
   GO TO 20
62 SIR=0.
   GO TO 20
   1 IF(G(3,2)) 2,5,2
   5 IF(G(1,2))95.5,24
95 SIR=0.
   COR=-1.
   GO TO 20
24 SIR=0.
   COR=1.
   GO TO 20
   2 RETA=ATANF(G(3,2) /G(1,2) )
   IF(RETA)85,85,89
89 IF(G(3,2))83,82,82
82 SIR=SINF(BETA)
   COR=COSF(BETA)
   GO TO 20
83 RETA=BETA+3.1415927
   GO TO 82
85 IF(G(3,2)) 82,82,83
20 C2D=G(2,2)*G(2,2)
   TAND=SQRTF(1.-C2D)/(-G(2,2) )
   TAN2=TAND*TAND
   XB=1./FG2/TAN2
   IF(225.-XB)6,6,7
   6 TAA=0.
   GO TO 63
   7 TAA=S*C2D*SQRTF(225.-XB)*(TAN2*EL*EL*.53333-50. +.03333/GM/GM/
   1TAN2)
63 CONTINUE
   CD=-G(2,2)
   SD=SQRTF(1.-C2D)
   DT=ATANF(TAND)
   X0=-15.
   FIN=0.
   IF (DT)8,9,9
   8 DT=DT+3.14159
   9 IF(G(2,2)) 10,11,11
11 TR=1.
   IF(1.97965-DT)12,12,13
12 X1=15.
   GO TO 14
13 X1=-.5/GM/TAND
14 H=(X1-X0)/EN
   DO 16 I=1,20.2
   AI=I
   F=X0+AI*H
   ASSIGN 17 TO M
   GO TO 30
17 FIN=FIN+4.*F
   F=F+H
   ASSIGN 18 TO M
   GO TO 30

```

```

18 FIN=FIN+2.*F
16 CONTINUE
   FIN=H*(FIN-F)/3.
   TR= 4.*S*FIN*TR
   GO TO 26
10 IF(DT-1.16194)15,15,19
15 TR=-1.
   GO TO 12
19 X1=-.5/GM/TAND
   X01=X1
   X11=15.
   FIN=0.
   H=(X1-X0)/EN
   DO 41 I=1,20,2
   AI=I
   F=X0+AI*H
   ASSIGN 42 TO M
   GO TO 30
42 FIN=FIN+4.*F
   F=F+H
   ASSIGN 43 TO M
   GO TO 30
43 FIN=FIN+2.*F
41 CONTINUE
   FIN=H*(FIN-F)/3.
   H=(X11-X01)/EN
   F=X01
   ASSIGN 21 TO M
   GO TO 40
21 GIN=F
   DO 44 I=1,20,2
   AI=I
   F=X01+AI*H
   ASSIGN 22 TO M
   GO TO 40
22 GIN=GIN+4.*F
   F=F+H
   ASSIGN 23 TO M
   GO TO 40
30 F=(2.*GM*E*SD+CD)**2
   X2=F*E
   XB=SQRTF(1./FG2+X2)
   XC=SQRTF(225.-X2)
   XD=ATANF(XC/XB)
   XA=(FG2/2.*E*X2-E*(TGL-1.))/(2.*GM)/SQRTF(1.+FG2*X2)*XD
   F=F*(XA+E/2.*(XC-XB*XD))
   GO TO M
40 X2=E*E
   XC=SQRTF(225.-X2)
   XB=SQRTF(1./FG2+X2)
   XD=ATANF(XC/XB)
   IF(15.-(E+1./GM/TAND))70,70,71
70 XF=0.
   GO TO 72

```

```

71 CONTINUE
  XF=SQRTF(225.-(E+1./GM/TAND)**2)/XB
72 CONTINUE
  XE=ATANF(XF)
  XA=(FG2/2.*E*X2-E*(TGL-1.))/(2.*GM)/SQRTF(1.+FG2*X2)*(XD-XE)
  F=(2.*GM*E*SD+CD)**2
  F=F*(XA+E/2.*(XC-XF*XB)*(XD-XE))
  GO TO M
23 GIN=GIN+2.*F
44 CONTINUE
  GIN=H*(GIN-F)/3.
  TR= 4.*S*(FIN-GIN)
26 TAR=.25*TAA+.75*TR
  T(1)=-SIB*TAB
  T(2)=0.
  T(3)=COB*TAB
  SP=SPBS
  SY=SYBS
  SR=SRBS
  CY=CYBS
  CP=SQRTF(1.-SP*SP)
  CR=SQRTF(1.-SR*SR)
  DO 27 NA=1,3
  DO 27 NB=1,3
27 G(NB,NA)=RA(NA,NB)
  DO 28 L=1,3
28 XI(L,L)=G(L,1)**2*ZI(1)+G(L,2)**2*ZI(2)+G(L,3)**2*ZI(3)
  XI(1,2)=G(1,1)*G(2,1)*ZI(1)+G(1,2)*G(2,2)*ZI(2)+G(1,3)*G(2,3)*
17I(3)
  XI(1,3)=G(1,1)*G(3,1)*ZI(1)+G(1,2)*G(3,2)*ZI(2)+G(1,3)*G(3,3)*
17I(3)
  XI(2,1)=XI(1,2)
  XI(3,1)=XI(1,3)
  XI(2,3)=G(2,1)*G(3,1)*ZI(1)+G(2,2)*G(3,2)*ZI(2)+G(2,3)*G(3,3)*
17I(3)
  XI(3,2)=XI(2,3)
  A1=SP*CY
  A2=-SP*SR*SY+CP*CR
  A3=-SP*CR*SY-CP*SR
  RA=XI(1,1)*A1+XI(1,2)*A2+XI(1,3)*A3
  RB=XI(2,1)*A1+XI(2,2)*A2+XI(2,3)*A3
  RC=XI(3,1)*A1+XI(3,2)*A2+XI(3,3)*A3
  T(1)=15.9E-9*(-BB*A3+BC*A2) +I(1)
  T(2)=15.9E-9*(BA*A3-BC*A1) +I(2)
  T(3)=15.9*(BB*A1-BA*A2)*1.E-9 +I(3)
  RETURN
  END

```

```

SUBROUTINE RK
  DIMENSION X(6),U(3),T(3),ZI(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(6)
  DIMENSION R(6),SA(6),SB(6),SC(6),SD(6)
  COMMON X,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,Z,S,D,NA,TI
  DO 2 N=1,6
2  R(N)=X(N)
  CALL DER
  DO 3 N=1,6
3  SA(N)=D(N)*DT
  DO 4 N=1,6
4  X(N)=R(N)+SA(N)/2.
  TI=TI+DT/2.
  CALL DER
  DO 5 N=1,6
5  SB(N)=D(N)*DT
  DO 6 N=1,6
6  X(N)=R(N)+SB(N)/2.
  CALL DER
  DO 7 N=1,6
7  SC(N)=D(N)*DT
  TI=TI+DT/2.
  DO 8 N=1,6
8  X(N)=R(N)+SC(N)
  CALL DER
  DO 9 N=1,6
  SD(N)=D(N)*DT
9  X(N)=R(N)+(SA(N)+SD(N))/6.+(SB(N)+SC(N))/3.
  RETURN
  END

```

## ELASTICITY MODEL

DATE 11 MAR 68

C  
C  
C  
C  
C  
C

```

C 7094      7094      7094      7094
C1 NA,NB,PA,PB,PC      215,5F10.5
C2 ZI(1-3), ZM(1-3), AL(1-3),      9F8.5
C3 ZL(1,1), (1,2), ..... (3,3)      9F8.5
C4 AX,RX,PIB,R0B,YB,PDBS,RDBS,YDBS      9F8.5
C5 Y0,AT,0NG,XLAT,XL0N,DT,AE      9F8.5
C6 T(1) T(2) T(3)      9F8.5
C XF PREVENTS JAGS COMING IN; AE INCREASES ZERO BAND;AX,BX LEAD FACTORS
C 7=0 USES TOR SUBROUTINE
C (3-6) (10-16) 20,23,27,30,31,35, (100-116)
      DIMENSIONX(10),U(3),T(3),ZI(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(20)
      DIMENSION RA(3,3),RB(3,3),RC(3,3),RD(3,3),ETA(3),C(3),NF(3)
      DIMENSION XX(10)
      COMMON /CON/C,ETA,NF,XF,AE,AX,BX,XX
      COMMON X,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI/HYB/PA,PB,PC,AC
      COMMON /P6SIN/XLAT,XL0N,PIR,R0R,PBS,YBS,RBS,PIB,YB,R0B
      COMMON /TEMP/T1,T2,T3,T4,T5,T6,T7,T8,T9,T0
13 TI=0.
12 CONTINUE
      I=0.
      T4=6.610808
      7K=T4
      NC=2
      RAD=.017453
      AC=1.
      DO 1 I=1,3
      NF(I)=1.
      C(I)=1.
1 CONTINUE
      READ (5,100)NA,NB,PA,PB,PC,Z
      READ(5,101) (ZI(I),I=1,3),(ZM(I),I=1,3),(AL(I),I=1,3)
      READ (5,101) ((ZL(I,J),J=1,3),I=1,3)
      READ (5,101) AX,BX,XF,PIB,R0B,YB,RDBS,YDBS,PDBS
      READ(5,101) Y0,AT,0NG,XLAT,XL0N,DT,AE
      READ (5,101) T, (D(I),I=1,14)
C D(1)-D(14) LENGTH, MASS, SPRING CONSTANT, DAMPING COEF.
      DO 35 I=1,3
      ZIM(I) =ZI(I)/ZM(I)
      U(I)=ZM(I)
35 CONTINUE
      IF(ZL(1,2).EQ.0..AND.ZL(1,3).EQ.0.) NC=1
23 T7=AT
      T8=0NG
      T9=XLAT
      T10=XL0N
      0NG=0NG-AL(3)
      XL0N=XL0N-AL(3)

```

```

3 AT=AT*RAD
  ANG=ANG*RAD
  Y0=Y0*RAD
  SAT=SIN(A1)
  S0N=SIN(ANG)
  CAT=COS(A1)
  C0N=COS(ANG)
  T1=SIN(Y0)
  T2=COS(Y0)
  T3=T4-CAT*C0N
  P2=ATAN(-CAT*S0N/T3)
  SP2=SIN(P2)
  R2=ATAN(SAT/(COS(P2)*T3-CAT*S0N*SP2) )
  T5=T1*COS(R2)
  YBS=ATAN(T5/SQRT(1.-T5**2))
  T5=SIN(R2)/COS(YBS)
  RBS=ATAN(T5/SQRT(1.-T5**2))
  T5=(SP2*T2-T1*SIN(R2)*COS(P2))/COS(YBS)
  PBS=ATAN(T5/SQRT(1.-T5**2))
  WRITE (6,105)
  WRITE (6,106)((ZL(I,J), J=1,3), I=1,3)
  WRITE (6,107)
  TEM=Y0/RAD
  WRITE (6,106)TEM , T7, T8, T9, T10, RBS, YBS, PBS, DT
  WRITE (6,108)
  WRITE (6,106) (Z1(I), I=1,3), (ZM(I), I=1,3), (AL(I), I=1,3)
  WRITE (6,109)
  WRITE (6,106)AX, BX, XF, AE
  CALL INIT
  DO 37 I=1,3
    XX(I)=X(I)
    XX(I+3)=0.
    X(I+6)=0.
37 XX(I+6)=0.
    XX(10)=0.
    X(10)=0.
    CALL STUP(0.,.35692,0.,RA)
    CALL STUP(AL(3),0.,0.,RB)
    DO 4 I=1,3
      DO 4 J=1,3
        RD(I,J)=0.
      DO 4 K=1,3
4 RD(I,J)=RD(I,J)+RA(I,K)*RB(K,J)
    CALL STUP(PIR,R0R,0.,RA)
    DO 5 I=1,3
      DO 5 J=1,3
        RB(J,I)=0.
      DO 5 K=1,3
5 RB(J,I)=RB(J,I)+RD(I,K)*RA(K,J)
C 5 TAKES THE TRASPOSE AS IT COMPUTES THE PRODUCT
    CALL STUP (PIB,R0B,YB,RA)
    SPIR=SIN(PIR)
    SR0R=SIN(R0R)
    CPIR=COS(PIR)

```

```

      CROR=COS(XSR)
      IF(7.NE.0.)          GO TO 16
6  CALL TOR (RA,RB,SPIR,SROR,CPIR,CROR)
15 CONTINUE
      T4=T7
      T5=T8
      T7=T(1)*PA/ZI(1)
      T8=T(2)*PA/ZI(2)
      T9=T(3)*PA/ZI(3)
      WRITE (6,110)
      WRITE (6,106) T(1),T(2),T(3),T7,T8,T9
      IF (NA)15,14,15
15 CALL OUT
      GO TO 10
14 WRITE (6,112)
      T1=X(1)*PB
      T2=X(2)*PB
      T3=X(3)*PB
      T4=X(4)*PC
      T5=X(5)*PC
      T6=X(6)*PC
      WRITE (6,106) T1,T1,T2,T3,T4,T5,T6
      WRITE (6,113)
      PAUSE 11
10 IF(7.EQ.0.) CALL TOR(RA,RB,SPIR,SROR,CPIR,CROR)
      DO 38 JB=1,10
38 XX(JB)=X(JB)
      CALL REG
      DO 11 J=1,NA
      CALL RK
      I=L+1
      ASSIGN 31 TO NG
      GO TO 30
31 CONTINUE
      IF(NF(1)+NF(2)+NF(3)-6) 27,20,27
27 IF(L.LT.NB)GO TO 11
      CALL OUT
      I=0
11 CONTINUE
      IF (SENSE SWITCH 1) 41,40
41 READ (101,100) NB
      GO TO 10
40 IF(SENSE SWITCH 2) 20,10
20 CALL OUT
      PAUSE 16
      GO TO 13
30 CONTINUE
C FOLLOWING COMPUTES GROUND TRACK, WHERE T4 IS LAT AND T5 IS LONG
      S1=SIN(X(3))
      S2=SIN(X(2))
      S3=SIN(X(1))
      C1=COS(X(3))
      C2=COS(X(2))
      C3=COS(X(1))

```

```

S7=SR0R*S1*C2+CR0R*S2
C7=SQRT(1.-S7*S7)
S8=(SR0R*(C1-C3-S1*S2+S3)+CR0R*S3*C2)/C7
C8=SQRT(1.-S8*S8)
S6=(SPIR*C1*C2+CPIR*CR0R*S1*C2-CPIR*SR0R*S2)/C7
C6=SQRT(1.-S6*S6)
F2=ATAN(S7/C7/C8)
T2=SIN(F2)
D2=COS(F2)
T1=S6*C7*D2+T2*(C6*S8+S6*C8*S7)
D1=SQRT(1.-T1*T1)
T3=S8*C7
D3=SQRT(1.-T3*T3)
Y1=T3*(ZK*D3*D1-SQRT(1.-ZK*ZK*(1.-D3*D3*D1*D1)))
Z1=SQRT(1.-Y1*Y1)
T4=ATAN(Y1/Z1)/RAD
A1=SQRT(Z1*Z1-ZK*ZK*T1*T1)
T5=ATAN(T1*(A1-ZK*D1)/(A1+ZK*T1*T1)/D1)/RAD +AL(3)
GO TO NG,(32,31)
100 FORMAT (2I5,5F10.5)
101 FORMAT (9F8.5)
102 FORMAT(46H TYPE POS YO,AT,ONG,TARGET LAT AND LONG 9F5.3)
103 FORMAT (9F5.3)
106 FORMAT (1P0E14.5)
105 FORMAT (1H0,2X 10HLAMDA(1,1),4X10HLAMDA(1,2),4X10HLAMDA(1,3),4X10H
11AMDA(2,1),4X10HLAMDA(2,2),4X10HLAMDA(2,3),4X10HLAMDA(3,1),4X10HLA
2MDA(3,2),4X10HLAMDA(3,3) )
107 FORMAT (1H0,4X2HY0,12X2HA1,11X3H0NG,11X3HLAT,11X3HL0N,11X3HRBS,11X
23HYRS,11X3HPBS,12X2HDT )
108 FORMAT (1H0,4X8HM0MENT 1,6X8HM0MENT 2,6X8HM0MENT 3,8X4HZM 1,10X4HZ
2M 2,10X4HZM 3,8X7HALPHA 1,7X7HALPHA 2,7X7HALPHA 3 )
109 FORMAT (1H0,5X2HAX,12X2HBX,12X2HXF,12X2HAE )
110 FORMAT (1H0,4X8HTORQUE 1,6X8HTORQUE 2,6X8HTORQUE 3,15X14HSCALED TO
1RQUES )
111 FORMAT (1H0, 5X 4HTIME, 10X4HX(1), 10X4HX(2), 10X 4HX(3), 10X
14HX(4), 10X 4HX(5), 10X 4HX(6) )
112 FORMAT (1H0,4X4HTIME, 19X 26HSCALED VALUES OF X(1-6) )
113 FORMAT (1H1 )
114 FORMAT (10HSET ANALOG )
115 FORMAT (20X 4HD(1), 10X 4HD(2), 10X 4HD(3), 10X 4HD(4),
110X 4HD(5), 10X 4HD(6) )
116 FORMAT (20X 4HU(1), 10X 4HU(2), 10X 4HU(3), 10X 3HX1S, 11X 3HX4S,
110X 1HS )
END

```



```

      SUBROUTINE REG
C 7094      7094      7094      7094
C (1-17),60,(63-66),68,70,73,(101-129),134,135,138
C ,200-203),(207-217),(222-225),345
      DIMENSION XX(17),U(3),T(3),ZI(3),ZM(3),ZIM(3),X0(10),ZL(3,3)
      DIMENSION AL(3),NF(3),ETA(3),U0(3),C(3),TIM(3),TS(3),D(20),XA(10)4
      COMMON XA,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI/HYB/PA,PB,PC,AC
      COMMON /CON/C,ETA,NF,XF,AE,AX,BX,XX
      COMMON /TEMP/T1,T2,T3,T4,T5,T6,T7,T8,T9,T0
      DO 115 I=1,3
200 U0(I) = U(I)
      GO TO (201,202,203),I
201 R=(T(1)+(ZI(2)-ZI(3))*XX(5)*XX(6)*AL(1))/ZM(1)
      T1=XX(4)-XX(2)*XX(6)*AL(1)
      GO TO 11
203 R= ( T(3)+(ZI(1)-ZI(2))*XX(4)*XX(5)*AL(1))/ZM(3)
      T1=XX(6)+XX(1)*XX(5)*AL(1)
      11 X=XX(I)*ZIM(I)*SIGN(1.,-R)
      Y=XX(I+3)*ZIM(I)*SIGN(1.,-B)
      GO TO 18
202 R=(T(2)+(ZI(3)-ZI(1))*XX(6)*XX(4)*AL(1))/ZM(2)
      T1=XX(5)-XX(1)*XX(6)*AL(1)
      X=XX(I)*ZIM(I)*SIGN(1., B)
      IF (B.EQ.0.) X=-X
      Y=XX(I+3)*ZIM(I)*SIGN(1.,-B)
      18 IF(X+Y)2,2,1
      1 U(I)=SIGN(ZM(I),-X)*SIGN(1.,-B)
      ETA(I)=-1.
      C(I)=1.
      GO TO 10
      2 T2=200.*DT
      IF(ZIM(1)*ABS(X0(I+3)-XX(I+3))-T2)12,12,13
      12 IF(ABS(X0(I)-XX(I))-T2*ABS(XX(I+3)))14,14,13
      13 ETA(I)=-1.
      C(I)=1.
      GO TO 9
      14 C(I)=C(I)*U(I)
      9 CONTINUE
      A GO TO(209,345),NC
C NC=1 TIME OPTIMAL,208 USES MIN,209 USES MAX.
      345 IF(C(I))208,209,208
      208 IF(ABS(XX(I+3))-ABS(T1))212,212,210
      209 IF(ABS(XX(I+3))-ABS(T1))210,212,212
      210 Y=SIGN(T1*ZIM(I),Y)
      212 IF(ETA(I))207,207,213
      213 Y=Y*XF
      207 T3=Y*Y/2.
      P2=X-T3/(B-1.)
      P4=X-T3/(B+1.)
      GO TO (63,60),NC
      63 IF(X)64,66,65
      64 IF(P2)17,16,16
      65 IF(P4)17,17,16

```

```

66 IF(Y) 17,15,16
60 IF(X) 68,66,70
68 IF(P2) 73,16,15
70 IF(P4) 17,17
73 CONTINUE
   T4=ABS(ZL(I,2)*H/ZL(I,3))
   R=SQRT(2.*T4)
   S=SQRT(ZL(I,1)/ZL(I,3)+T4)
   IF(X) 3,66,5
   3 IF(Y-S) 19,4,4
10 IF(R) 26,25,26
25 QX=-10000.
   GO TO 20
26 RA=-ABS(B)
   QX=(BA*S*S-2.*P*S+R*P)/(2.*BA*(BA-1.))
20 IF(X-QX) 17,17,21
21 G=-7L(I,1)-ZL(I,3)*Y**2+7L(I,2)*PA
   7=-2.*ZL(I,2)+ZL(I,3)*(Y**2-2.*(BA-1.)*X )
   IF((-BA*Z*Z+2.*G*Z +G*G)*Y*Y-2.*G*G*(BA-1.)*X) 17,15,15
   4 IF(Y-S-R-AE) 15,15,16
   5 IF(Y-(R-S)) 6,15,22
22 IF(R) 27,28,27
28 QX=10000.
   GO TO 23
27 RA=-ABS(B)
   QX=(BA*(R-S)**2-2.*R*S+R*R)/(2.*BA*(BA+1.))
23 IF(X-QX) 24,16,16
24 G=-7L(I,1)-ZL(I,3)*Y**2-ZL(I,2)*BA
   7=-2.*ZL(I,2)+ZL(I,3)*(Y**2-2.*(BA+1.)*X )
   IF((-BA*Z*Z+2.*G*Z +G*G)*Y*Y-2.*G*G*(BA+1.)*X) 16,15,15
   6 IF(Y-(-S-AE)) 17,15,15
15 U(I)=0.
   GO TO 7
16 U(I)=SIGN(ZM(I),B)
   IF(B.EQ.0.) U(I)=-U(I)
   GO TO 7
17 U(I)=SIGN(ZM(I),-B)
   7 IF( 1-2) 31,32,31
32 FTA(I)=U(I)*X
   GO TO 10
31 FTA(I)=U(I)*XX(I)
10 X0(I)=XX(I)
   X0(I+3)=XX(I+3)
C 123 REGINS TIME SYNCH
123 X=XX(I)*ZIM(I)
   IF (1.EQ.2) X=-X
   Y=XX(I+3)*ZIM(I)
   T6=U(I)
121 T5=Y*Y/2.
   P2=X-T5
   P4=X+T5
   GO TO(117,101),NC
117 IF(Y) 107,107,109
107 IF(P2) 110,110,111

```

```

111 X=-X
    Y=-Y
    T6=-U(I)
    GO TO 121
109 IF(P4) 112,111,111
110 TS(I)=-1.
    GO TO 113
112 TS(I)=ABS(Y/2.)-X/Y
    IF(T6*Y.LE. 0.) TS(I)=-1.
113 TIM(I)=2.*SQRT(ABS(P2))-Y
    GO TO 211
101 PS=X+(.5+2.*ZL(I,2)/(ZL(I,1)-ZL(I,3)*Y**2))*Y**2
    P=Y*ABS(Y)+2.*X
    S=SQRT(ZL(I,1)/ZL(I,3))
    IF(P) 87,79,111
87 IF(Y-S)74,89,80
89 TS(I)=-1.
    TIM(I)=Y-P4/S
    GO TO 211
74 IF(PS)75,75,76
75 TQ=Y**2-2.*X
    TA=2.*ZL(I,1)+4.*ZL(I,2)
    TQ=SQRT((TA+ZL(I,3)*TQ)**2-8.*ZL(I,1)*ZL(I,3)*TQ)
    TQ=SQRT((TA+ZL(I,3)*TQ-TQ)/4./ZL(I,3))
C TO IS OMEGA S1
    TIM(I)=TQ-P2/TQ-Y
C REGION I
    IF(Y)77,77,78
77 TS(I)=-1.
    GO TO 211
78 TS(I)=Y/2.-X/Y
    IF(T6*Y.LE. 0.) TS(I)=-1.
    GO TO 211
76 IF (Y) 75,75,79
79 TIM(I)=      Y/2.-X/Y
    GO TO 77
84 X=-X
    Y=-Y
    GO TO 121
211 IF(ABS(XX(I))-.0020) 222,223,223
222 IF(ABS(XX(I+3))-.00005) 224,223,223
224 U(I)=0.
    TIM(I)=0.
    NF(I)=2
    GO TO 115
223 K=NF(I)
    GO TO (115,216).K
216 IF(ABS(XX(I))-.005)226 ,225,225
226 U(I)=0.
    TIM(I)=0.
    GO T O 115
225 NF(I)=1
115 CONTINUE
127 TMAX=AMAX1(TIM(1),TIM(2),TIM(3))*AX

```

129 DO 124 J=1,3  
K=NF(J)  
128 GO TO (138,124),K  
138 IF(TS(J))124,125,125  
125 IF(TMAX-TS(J))124,124,134  
134 IF(TMAX-TS(J)-BX)135,126,126  
135 IF(U0(J))124,126,124  
126 U(J)=0.  
124 CONTINUE  
RETURN  
END

```

SUBROUTINE DER
  DIMENSION X(10), U(3), T(3), ZI(3), ZM(3), ZIM(3), AL(3), ZL(3,3), D(20)
  COMMON X, U, T, ZI, ZM, ZIM, AL, ZL, DT, NC, R, S, D, NA, TI
  COMMON /TEMP/ T1, T2, T3, T4, T5, T6, T7, T8, T9, T0
  EQUIVALENCE (XL, D(11)), (XM, D(12)), (ALF, D(13)), (BET, D(14))
C  THIS PROGRAM IS ONLY TO CHECK ELASTIC EFFECTS
C  XL IS ROOM LENGTH. XM IS MASS (SLUGS)
C  ALF IS SPRING CONSTANT8 BET IS DAMPING
  ZI(1)=ZI(1)-XM*XL**2
  ZI(3)=ZI(3)-XM*XL**2
  T7=X(4)*X(7)+X(5)*XL      +X(6)*X(8)
  T8=X(4)**2+X(5)**2+X(6)**2
  T9= XM*(-2.*(X(6)*X(9)-X(4)*X(10))-X(5)*T7+XL*T8-(D(6)*X(7)-D(4)D
1*X(8)))
  T1=XL*(ALF*X(8)+BET*X(10))-X(8)*T9
  T2=BET*(X(8)*X(9)-X(7)*X(10))
  T3=-XL*(ALF*X(7)+BET*X(9))+X(7)*T9
  N(1)=-AL(1)* X(2)*X(6) + X(4)
  N(2)= AL(1) * X(1) *X(6) - X(5)
  N(3)= AL(1)*X(1)*X(5) + X(6)
  N(4)= (AL(1) *(ZI(2)-ZI(3))* X(5)* X(6) + U(1) + T(1))/ZI(1)
  N(5)= (AL(1) *(ZI(3) -ZI(1))*X(4)*X(6) +U(2) +T(2))/ZI(2)
  N(6)= (AL(1)* (ZI(1)-ZI(2))* X(4) * X(5) + U(3) + T(3))/ZI(3)
  N(4)=D(4)+T1/ZI(1)
  N(5)=D(5)+T2/ZI(2)
  N(6)=D(6)+T3/ZI(3)
  N(7)=X(9)
  N(8)=X(10)
  N(9)=-(ALF*X(7)+BET*X(9))/XM-X(4)*T7+X(7)*T8-(D(5)*X(8)-D(6)*XL)
1-2.*X(5)*X(10)
  N(10)=-(ALF*X(8)+BET*X(10))/XM-X(6)*T7+X(8)*T8-(D(4)*XL-D(5)*X(7))
1+2.*X(5)*X(9)
  ZI(1)=ZI(1)+XM*XL**2
  ZI(3)=ZI(3)+XM*XL**2
  RETURN
END

```

```

SUBROUTINE RK
  DIMENSION X(10), U(3), T(3), ZI(3), ZM(3), ZIM(3), AL(3), ZL(3,3), D(20)
  DIMENSION R(10), SA(10), SB(10), SC(10), SD(10)
  COMMON X, U, T, ZI, ZM, ZIM, AL, ZL, DT, NC, Z, S, D, NA, TI
  DO 2 N=1,10
2  R(N)=X(N)
  CALL DER
  DO 3 N=1,10
3  SA(N)=D(N)*DT
  DO 4 N=1,10
4  X(N)=R(N)+SA(N)/2.
  TI=TI+DT/2.
  CALL DER
  DO 5 N=1,10
5  SB(N)=D(N)*DT
  DO 6 N=1,10
6  X(N)=R(N)+SB(N)/2.
  CALL DER
  DO 7 N=1,10
7  SC(N)=D(N)*DT
  TI=TI+DT/2.
  DO 8 N=1,10
8  X(N)=R(N)+SC(N)
  CALL DER
  DO 9 N=1,10
  SD(N)=D(N)*DT
9  X(N)=R(N)+(SA(N)+SD(N))/6.+(SB(N)+SC(N))/3.
  RETURN
  FND

```

```
SUBROUTINE DT0A  
  DIMENSION X(10), U(3), T(3), Z1(3), ZM(3), ZIM(3), AL(3), ZL(3,3), D(20)  
  COMMON X,U,T,Z1,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI/HYB/PA,PB,PC,AC  
  AC=AC  
  RETURN  
  END
```

```
SUBROUTINE AT0D  
DIMENSIONX(10),U(3),T(3),ZI(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(20)  
COMMON X,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI/HYB/PA,PB,PC,AC  
AC=AC  
RETURN  
END
```



```
SUBROUTINE CUL  
  DIMENSIONX(10),U(3),T(3),Z1(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(20)  
  COMMON X,U,T,Z1,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI  
  COMMON /TEMP/ T1,T2,T3,T4,T5,T6,T7,T8,T9,T0  
  WRITE (6,112) TI,X  
  WRITE (6,113) D  
  WRITE (6,113) U,T4,T5  
112 FORMAT (1F0, F6.2, 1P10E12.3)  
113 FORMAT (7X, 1P10E12.3)  
  RETURN  
  END
```

```

SURROUTINE STUP (CP,CR,CY,CA)
DIMENSION CA(3,3)
SINR=SIN(CR)
SINR=SIN(CR )
SINP=SIN(CP)
SINY=SIN(Y)
COSR=COS(CR)
COSP=COS(P)
COSY=COS(Y)
C WF COMPUTE THE MATRIX I+B
CA(1,1)=COSP*COSY
CA(1,2)=-COSP*SINR*SINY-SINP*COSR
CA(1,3)=-COSP*COSR*SINY+SINP*SINR
CA(2,1)=SINP*COSY
CA(2,2)=-SINP*SINR*SINY+COSP*COSR
CA(2,3)=-SINP*COSR*SINY-COSP*SINR
CA(3,1)=SINY
CA(3,2)=SINR*COSY
CA(3,3)=COSR*COSY
RETURN
END

```

```
SUBROUTINE TOR(RA,RB,SPIR,SROR,CPIR,CROR)
DIMENSIONX(10),U(3),T(3),ZI(3),ZM(3),ZIM(3),AL(3),ZL(3,3),D(20)
DIMENSION RA(3,3),RB(3,3),PC(3,3),PD(3,3),G(3,3),XI(3,3)
COMMON X,U,T,ZI,ZM,ZIM,AL,ZL,DT,NC,R,S,D,NA,TI
T(1)=1.E-7
T(2)=1.E-7
T(3)=1.E-7
RETURN
END
```

```

SUBROUTINE INIT
DIMENSION X(10), U(3), T(3), ZI(3), ZM(3), ZIM(3), AL(3), ZL(3,3), D(20)
DIMENSION A(3,3), B(3,3), C(3,3), H(3), W(3)
COMMON X, U, T, ZI, ZM, ZIM, AL, ZL, DI, NC, R, S, D, NA, TI
COMMON /POSIN/ XLAT, XLON, PIR, ROR, PBS, YBS, RBS, PIB, YB, ROB
XLON = XLON * 0.017453
XLAT = XLAT * 0.017453
SILAT = SIN(XLAT)
SILON = SIN(XLON)
COLAT = COS(XLAT)
COLON = COS(XLON)
CONST = 6.610808
PITCH = -ATAN ((+SILON * COLAT)/(CONST - COLON * COLAT))
COP1 = COS(PITCH)
SINP1 = SIN(PITCH)
ROLL = ATAN (SILAT/((CONST - COLON*COLAT)* COP1 - SILON*COLAT *
1 SINP1))
PIR = PITCH
ROR = ROLL
CRBS = COS(RBS)
SRBS = SIN(RBS)
CYBS = COS(YBS)
SYBS = SIN(YBS)
CPIB = COS(PIB)
CYB = COS(YB)
CROR = COS(ROB)
SYR = SIN(YB)
SROR = SIN(ROB)
SPIB = SIN(PIB)
R(1,1) = CPIB * CYB
R(1,2) = -CPIB * SROR * SYB - SPIB * CROR
R(1,3) = -CPIB * CROR * SYR + SPIB * SROR
R(2,1) = SPIB * CYB
R(2,2) = -SPIB * SROR * SYB + CPIB * CROR
R(2,3) = -SPIB * CROR * SYR - CPIB * SROR
R(3,1) = SYB
R(3,2) = SROR * CYB
R(3,3) = CROR * CYB
A(1,1) = 1.
A(1,2) = 0.
A(1,3) = SYBS
A(2,1) = 0.
A(2,2) = -CRBS
A(2,3) = CYBS * SRBS
A(3,1) = 0.
A(3,2) = SRBS
A(3,3) = CYBS * CRBS
DO 17 K = 1,3
DO 17 I = 1,3
C(I,K) = 0.
DO 17 J = 1,3
C(I,K) = B(I,J) * A(J,K) + C(I,K)
17 CONTINUE

```

```

      H(1)=RDBS
      H(2)=YBBS
      H(3)=PDBS
      DO 61 I = 1,3
      W(I) = 0.
      DO 62 J = 1,3
      W(I)=W(I)+C(I,J)*H(J)
62  CONTINUE
      X(I+3) = W(I)
61  CONTINUE
      SPIR=SIN(PIR)
      SROR=SIN(ROR)
      CPIR=COS(PIR)
      CROR=COS(ROR)
      SPRB=SIN(PIR-PBS)
      CPRB=COS(PIR-PBS)
      G1 = SROR * CYBS * SPRB + CROR * SYBS
78  FORMAT (F9.5)
      G2 = SQRT (1.-G1**2)
      X(2) = ATAN (G1/G2)
      G3=(-CROR*CYBS*SPRB+SROR*SYBS)/G2
      G4=SQRT (1.-G3*G3)
      X(3)=ATAN (G3/G4)
      G6 =(-SROR * CRBS * CPRB - SROR* SRBS *SYBS *SPRB + CROR*SRBS *
1CYBS)/G2
      G7= SQRT (1.- G6**2)
      X(1)= ATAN (G6/G7)
      RETURN
      END

```

## 7.5 Simulation Model For Analog Computer

### Equations of Motion

The equations of motion for the satellite rotational dynamics are written

$$I_1 \dot{\omega}_1 = (I_2 - I_3) \omega_2 \omega_3 + B_1 + T_1 \quad (7.5.1)$$

$$I_2 \dot{\omega}_2 = (I_3 - I_1) \omega_1 \omega_3 + B_2 + T_2 \quad (7.5.2)$$

$$I_3 \dot{\omega}_3 = (I_1 - I_2) \omega_1 \omega_2 + B_3 + T_3 \quad (7.5.3)$$

where  $B_i$  and  $T_i$  represent lumped control and disturbance torques respectively.

The Euler rate transformation equations for the rotational sequence  $\theta, -\psi, \phi$  (pitch, -yaw, roll) are written

$$\dot{\phi} = \omega_1 - \omega_3 \psi \quad (7.5.4)$$

$$\dot{\psi} = -\omega_2 + \omega_3 \phi \quad (7.5.5)$$

$$\dot{\theta} = \omega_3 + \omega_2 \phi \quad (7.5.6)$$

when small angle approximations ( $\cos x=1, \sin x=x$ ) are used for the angles  $\theta, -\psi$ , and  $\phi$ .

### Lumped Control Torque

For the case of no thruster misalignment the lumped control torques  $B_1, B_2$ , and  $B_3$  are written

$$B_1 = (t_1 \times d_1) u_1 \quad (7.5.7)$$

$$B_2 = (t_2 \times d_2) u_2 \quad (7.5.8)$$

$$B_3 = (t_3 \times d_3) u_3 \quad (7.5.9)$$

where,

$$u_i = \begin{cases} 1 (+) \text{ thrust} \\ 0 \text{ no thrust} \\ -1 (-) \text{ thrust} \end{cases}$$

$$t_i = \text{thrust}$$

$$d_i = \text{moment arm.}$$

When each of the six attitude control jets ( $\pm$  pitch,  $\pm$  yaw,  $\pm$  roll) can be misaligned in either azimuth ( $\gamma$ ) or elevation ( $\epsilon$ ) the equations for  $B_i$  must be rewritten.

$$\begin{bmatrix} B_1 \\ B_2 \\ B_3 \end{bmatrix} = \begin{bmatrix} b_{11}^+ & b_{12}^+ & b_{31}^+ \\ b_{21}^+ & b_{22}^+ & b_{32}^+ \\ b_{31}^+ & b_{32}^+ & b_{33}^+ \end{bmatrix} \begin{bmatrix} c_1 u_1 \\ c_2 u_2 \\ c_3 u_3 \end{bmatrix} + \begin{bmatrix} b_{11}^- & b_{12}^- & b_{31}^- \\ b_{21}^- & b_{22}^- & b_{32}^- \\ b_{31}^- & b_{32}^- & b_{33}^- \end{bmatrix} \begin{bmatrix} c_1 u_1 \\ c_2 u_2 \\ c_3 u_3 \end{bmatrix} \quad 7.5.10$$

where,

$$c_i = t_i \times d_i$$

$$b_{ij}^+ = \text{misalignment coefficients for (+) thrusters}$$

$$b_{ij}^- = \text{misalignment coefficients for (-) thrusters}$$

Equation (10) can be expanded to

$$B_1 = c_1 u_1 (b_{11}^+ \oplus b_{11}^-) + c_2 u_2 (b_{12}^+ \oplus b_{12}^-) + c_3 u_3 (b_{13}^+ \oplus b_{13}^-) \quad (7.5.11)$$

$$B_2 = c_1 u_1 (b_{21}^+ \oplus b_{21}^-) + c_2 u_2 (b_{22}^+ \oplus b_{22}^-) + c_3 u_3 (b_{23}^+ \oplus b_{23}^-) \quad (7.5.12)$$

$$B_3 = c_1 u_1 (b_{31}^+ \oplus b_{31}^-) + c_2 u_2 (b_{32}^+ \oplus b_{32}^-) + c_3 u_3 (b_{33}^+ \oplus b_{33}^-) \quad (7.5.13)$$

where  $(b_{ij}^+ + b_{ij}^-)$  is the misalignment coefficient for  $u = +1$  or  $(\oplus)$

$u = -1$ .

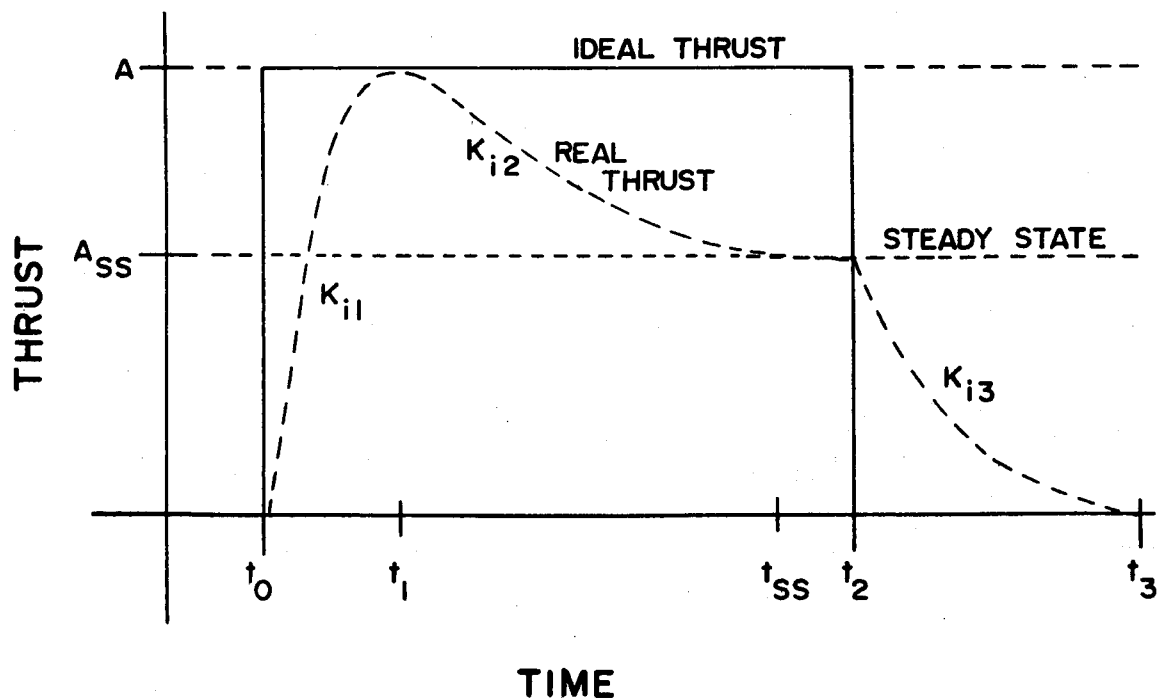
### Lumped Disturbance Torque

The lumped disturbance torques  $T_i$  represent total disturbance acting on the satellite. In general this is the sum of the gravity gradient torque and solar pressure torques.

### Thruster Real Effects (TRE)

The simulation of the Thruster Real Effects requires three separate time constants for each of the three axes. The time constants are designated  $k_{ij}$  where "i" is the axis.  $j=1$  is the rise time constant;  $j=2$  is the steady state decay time constant;  $j=3$  is the fall time constant.

The diagram below shows a typical curve used for the TRE simulation.





### Fuel Consumption

The fuel consumption is determined from the following equation

$$FC = \frac{1}{ISP} \int |u_1 t_1| + |u_2 t_2| + |u_3 t_3| dt.$$

This equation can be rewritten and scaled as follows

$$[10FC] = \sum_{i=1}^3 \left[ \frac{100I_i}{ISP \times d_i \times 5 \times 10^4} \right] \int |[5B_i \times 10^3 / I_i]| dt.$$

### Scaled Equations of Motion

$$[5\dot{\omega}_1 \times 10^3] = \frac{.8A_1}{10} [250\omega_2][250\omega_3] + (5B_1 \times 10^3 / I_1) + (5T_1 \times 10^3 / I_1)$$

$$[5\dot{\omega}_2 \times 10^3] = \frac{.8A_2}{10} [250\omega_1][250\omega_3] + (5B_2 \times 10^3 / I_2) + (5T_2 \times 10^3 / I_2)$$

$$[5\dot{\omega}_3 \times 10^3] = \frac{.8A_3}{10} [250\omega_1][250\omega_2] + (5B_3 \times 10^3 / I_3) + (5T_3 \times 10^3 / I_3)$$

$$[250\dot{\phi}] = [250\omega_1] - \frac{1}{5} [250\omega_3][5\psi]$$

$$[250\dot{\psi}] = -[250\omega_2] + \frac{1}{5} [250\omega_3][5\phi]$$

$$[250\dot{\theta}] = [250\omega_3] + \frac{1}{5} [250\omega_2][5\phi]$$

where,

$$A_1 = \frac{I_2 - I_3}{I_1}$$

$$A_2 = \frac{I_3 - I_1}{I_2}$$

$$A_3 = \frac{I_1 - I_2}{I_3}$$

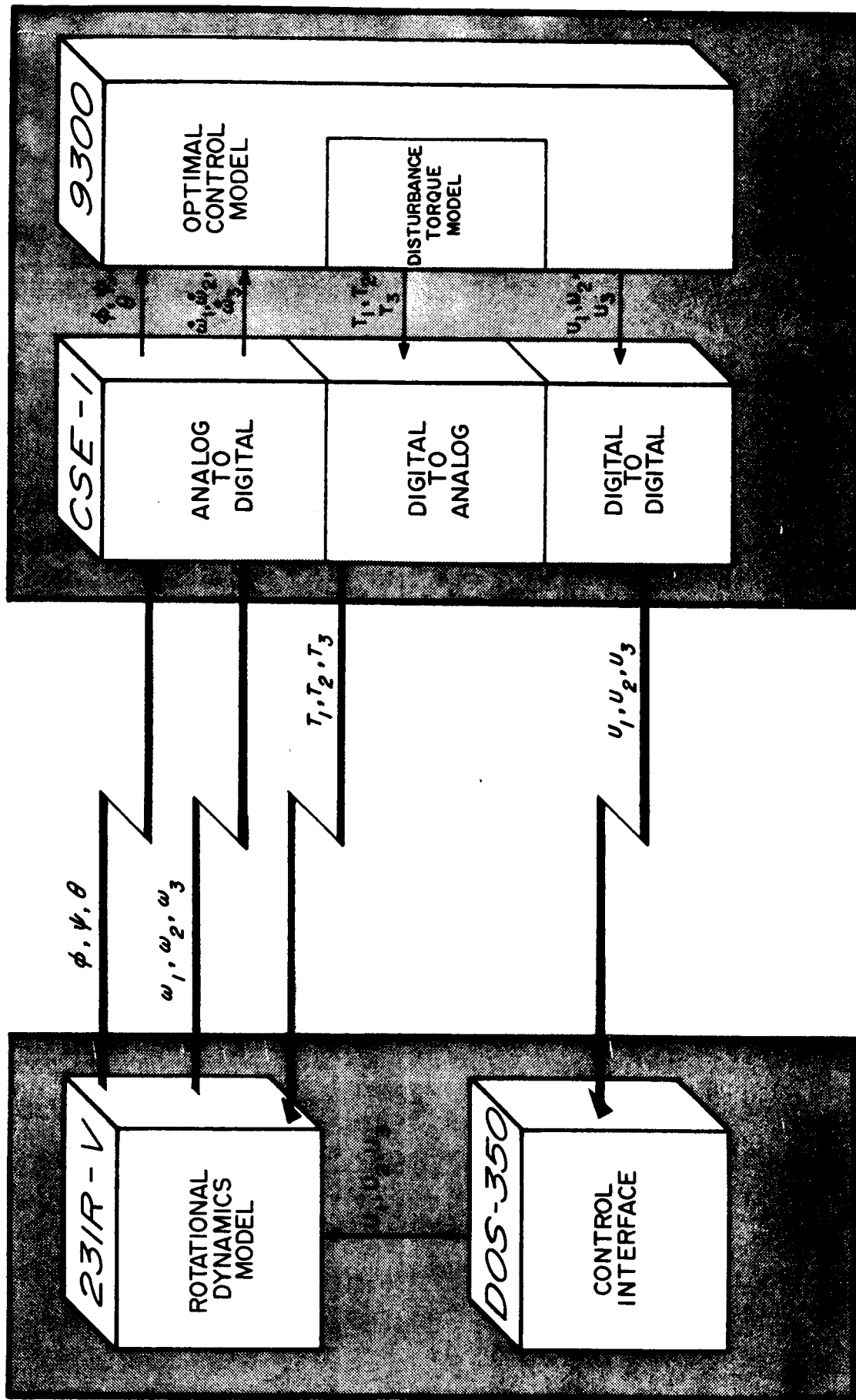
## 7.6 Overall Hybrid Simulation

The objective of the Hybrid Simulation was to gather performance and design data for the optimal attitude slew controller discussed in Parts I thru VI of this report. In order to accomplish this objective a Hybrid Computer system was used to model the Optimal Control Equations, Disturbance Torques, Satellite Rotational Dynamics, and Thrustor Real Effects. Each of these models were tied together as a total hybrid simulation by a Digital/Analog Interface.

The optimal control equations and external disturbance torques (Solar & Gravity Gradient) were programmed on the SDS 9300 digital computer. Position and rate information required in the optimal control equation were generated from the satellite rotational dynamics model programmed on the EAI 231R-V analog computer. The thrustor real effects were incorporated into the analog dynamics model.

The digital interface (CSE-1) and analog interface (DOS-350) made-up the linkage between the analog and digital computers. The digital interface provided A/D conversion, D/A conversion and timing. The control commands to the dynamics model were interfaced through the analog interface.

A block diagram of the simulation hardware and its function is shown on the next page. The symbols in this diagram are defined in Section 7.5.



SIMULATION BLOCK DIAGRAM

## 7.7 *Simulation Setup Procedures*

- a. Load digital program using standard 9300 procedures.  
When loading is complete computer will PAUSE.
- b. Place the data deck in the card reader and clear the  
PAUSE. Initial conditions for both the analog and digital  
computers will be output on the line printer and computer  
will PAUSE.
- c. Set analog potentiometers for the dynamics constants. (ref. 7.8)
- d. Set interval timers for the desired sampling time.
- e. Set up strip chart and X-Y recorders.
- f. Place analog computer in IC mode.
- g. Set analog potentiometers and switches in accordance with  
the printout. (ref. 7.8)
- h. Clear digital PAUSE.
- i. Place analog computer into operate. Run will now start.

## 7.8 *Analog Patching*

This section is divided into four parts.

- A. Initial Condition Setup
- B. Dynamics & Control
- C. Thrustor Misalignments
- D. Thrustor Real Effects & Fuel Consumption

INITIAL CONDITION SETUP

| VARIABLE |                | POTENTIOMETER |
|----------|----------------|---------------|
| MSG      | REPORT         |               |
| X 1 (0)  | $\phi$ (0)     | P 50          |
| X 2 (0)  | $\psi$ (0)     | P 51          |
| X 3 (0)  | $\theta$ (0)   | P 54          |
| X 4 (0)  | $\omega_1$ (0) | P 55          |
| X 5 (0)  | $\omega_2$ (0) | P 58          |
| X 6 (0)  | $\omega_3$ (0) | P 59          |

| <div>SWITCH</div> <div>VARIABLE</div> | 00 |   | 10 |   | 20 |   | 01 |   | 11 |   | 21 |   |
|---------------------------------------|----|---|----|---|----|---|----|---|----|---|----|---|
|                                       | R  | L | R  | L | R  | L | R  | L | R  | L | R  | L |
| X 1 (0)                               | -  | + |    |   |    |   |    |   |    |   |    |   |
| X 2 (0)                               |    |   | -  | + |    |   |    |   |    |   |    |   |
| X 3 (0)                               |    |   |    |   | -  | + |    |   |    |   |    |   |
| X 4 (0)                               |    |   |    |   |    |   | -  | + |    |   |    |   |
| X 5 (0)                               |    |   |    |   |    |   |    |   | -  | + |    |   |
| X 6 (0)                               |    |   |    |   |    |   |    |   |    |   | -  | + |

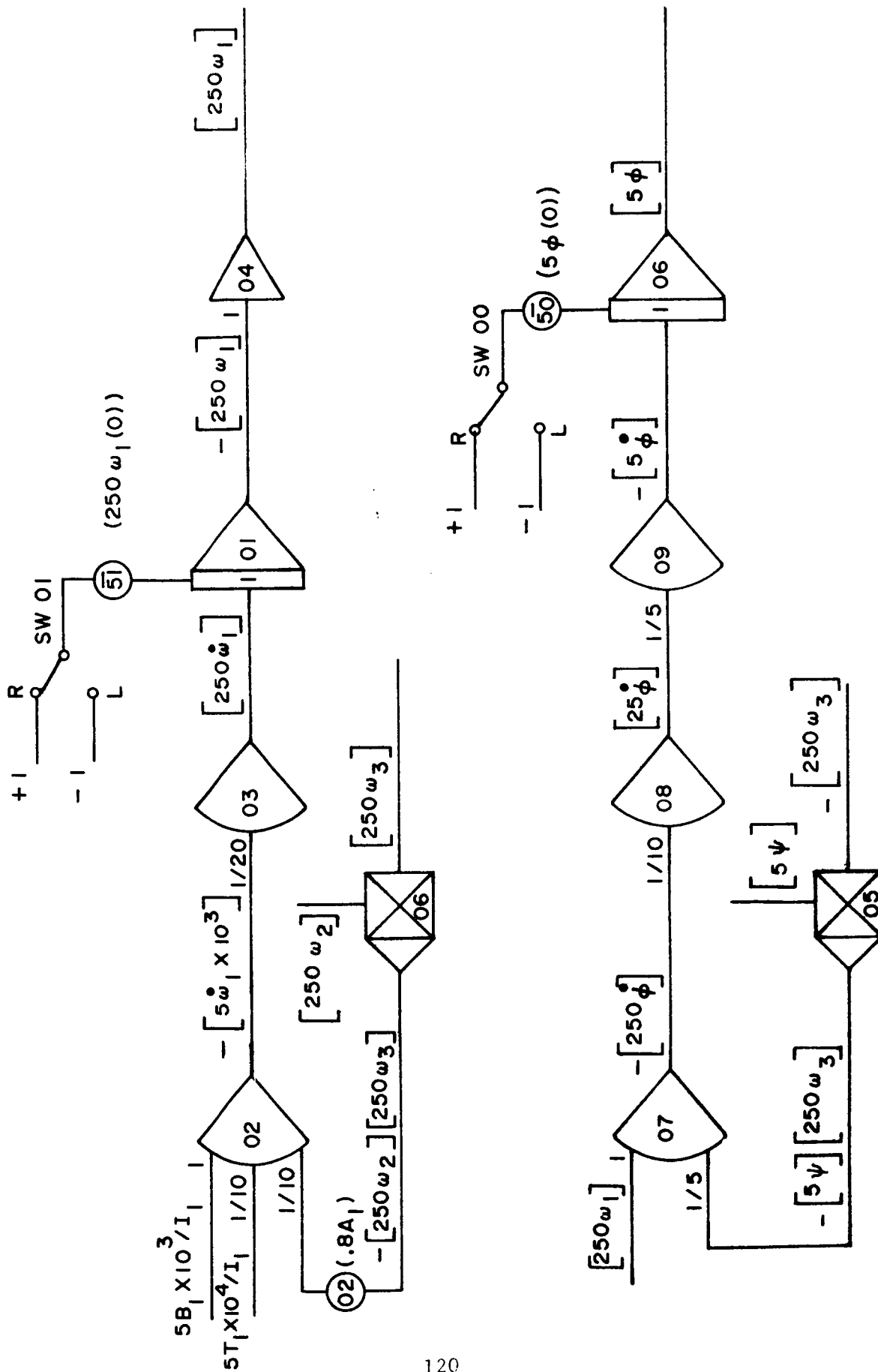
INITIAL CONDITION SET-UP

DYNAMICS & CONTROL



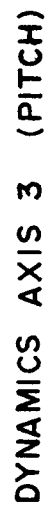
| POTENTIOMETER | VARIABLE                | VALUE |
|---------------|-------------------------|-------|
| Q06           | $(t_1 X d_1) (5)$       | .4000 |
| Q16           | $(t_2 X d_2) (5)$       | .2000 |
| Q26           | $(t_3 X d_3) (5)$       | .6000 |
| P06           | $1000 / I_1$            | .2793 |
| P16           | $1000 / I_2$            | .5076 |
| P26           | $1000 / I_3$            | .5000 |
| Q02           | $(.8)(I_2 - I_3) / I_1$ | .0067 |
| Q12           | $(.8)(I_3 - I_1) / I_2$ | .6416 |
| Q22           | $(.8)(I_1 - I_2) / I_3$ | .6440 |

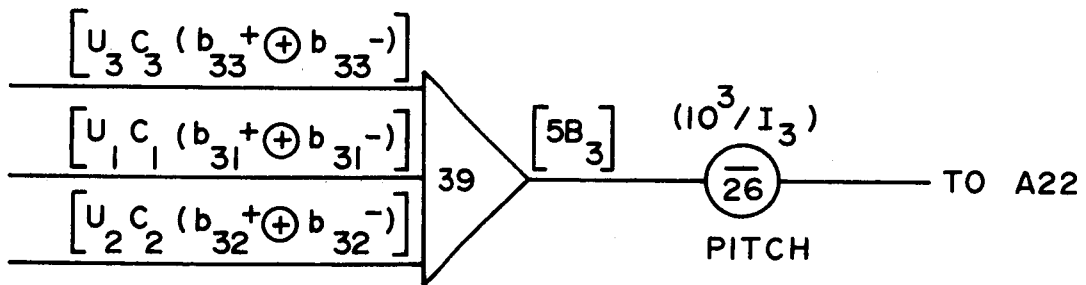
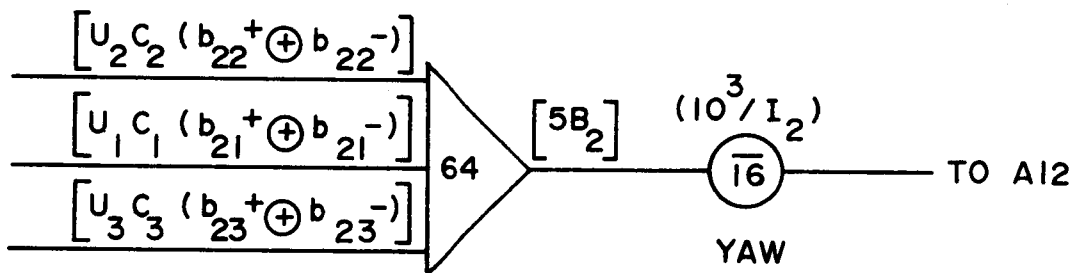
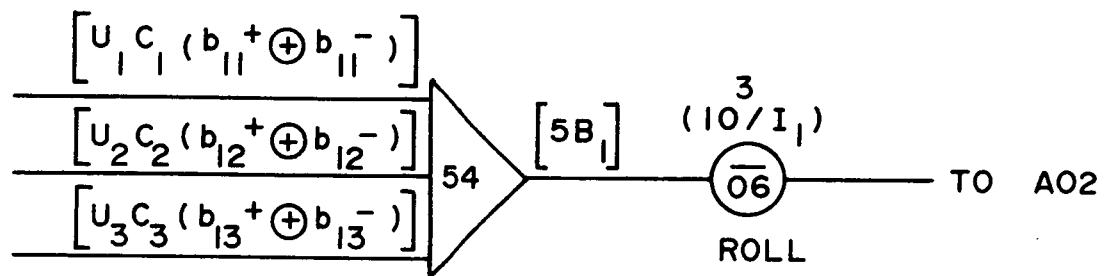
#### POTENTIOMETER SET-UP (DYNAMICS)



DYNAMICS AXIS 1 (ROLL)

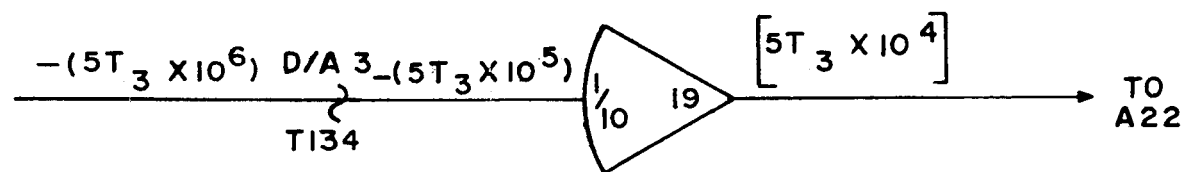
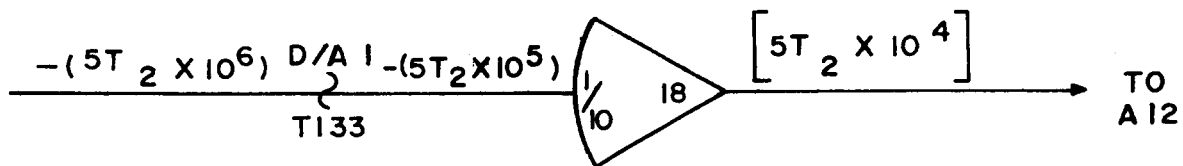
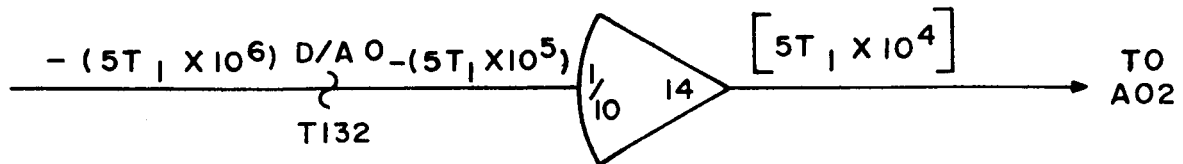




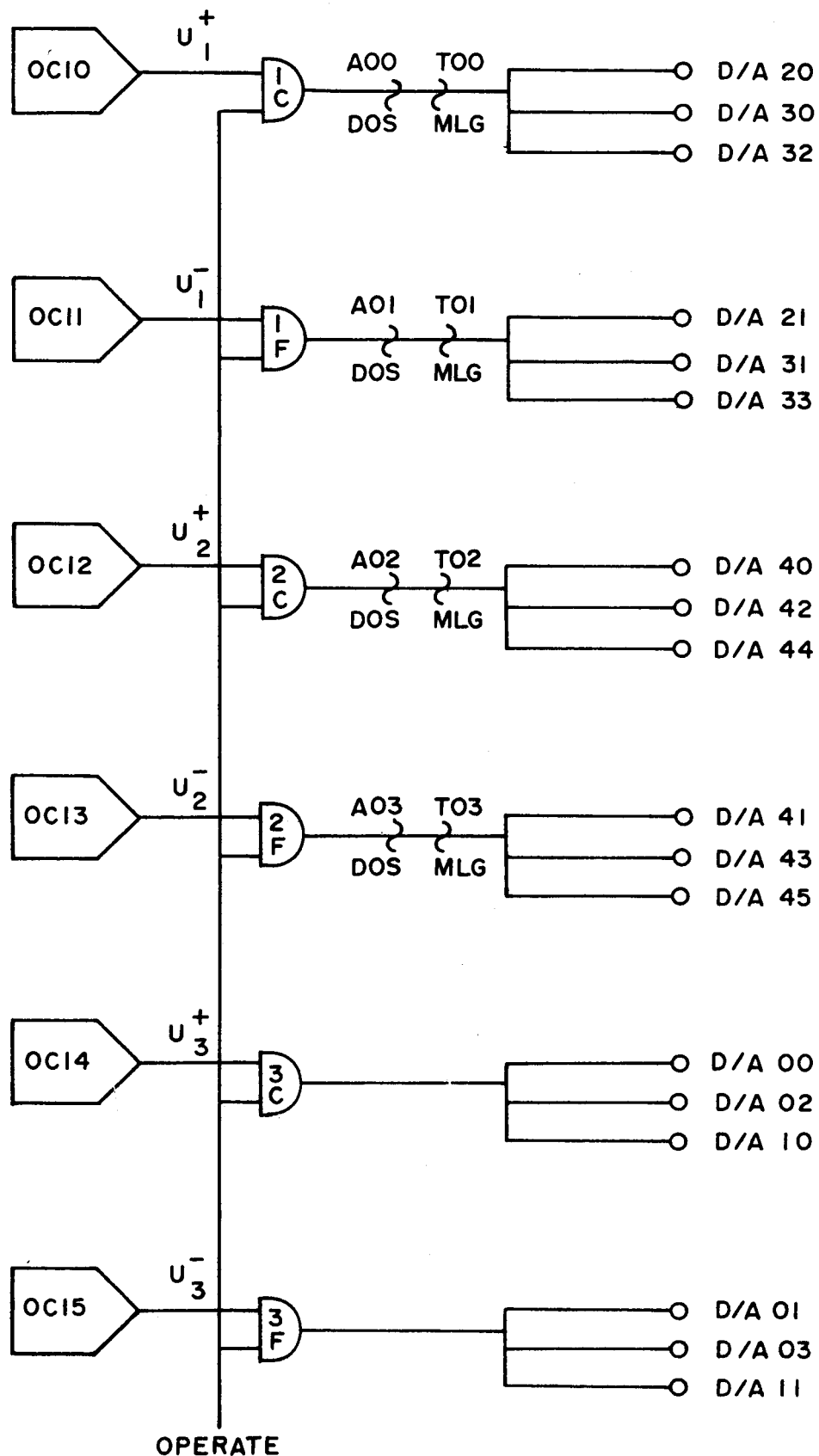


LUMPED THRUST TORQUE

FROM  
9300



DISTURBANCE TORQUES



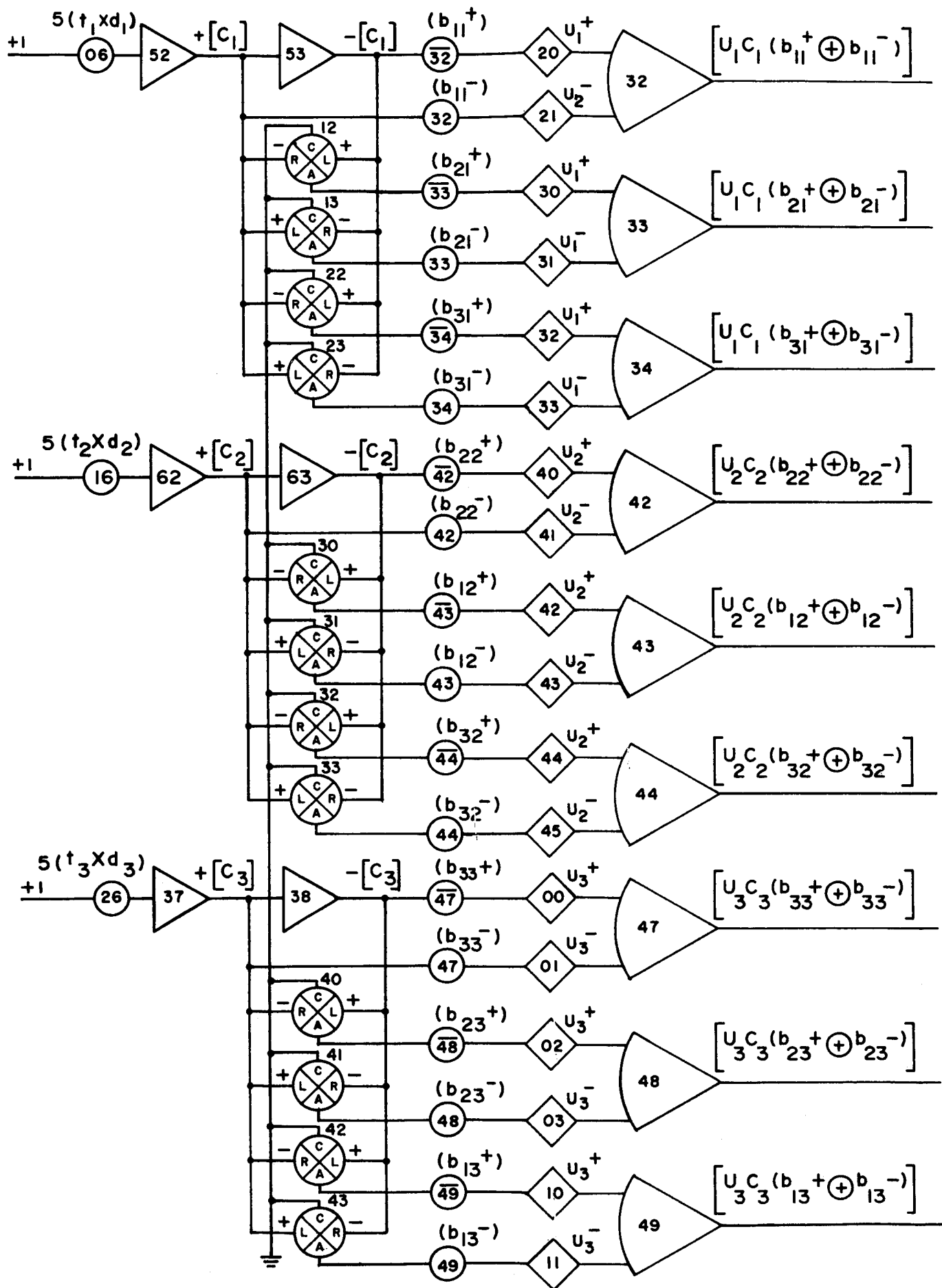
# THRUST CONTROL LOGIC

THRUSTOR MISALIGNMENTS



| POTENTIOMETER | VARIABLE     | VALUE |
|---------------|--------------|-------|
| P32           | $b_{11}^{+}$ |       |
| P33           | $b_{21}^{+}$ |       |
| P34           | $b_{31}^{+}$ |       |
| Q32           | $b_{11}^{-}$ |       |
| Q33           | $b_{21}^{-}$ |       |
| Q34           | $b_{31}^{-}$ |       |
| P42           | $b_{22}^{+}$ |       |
| P43           | $b_{12}^{+}$ |       |
| P44           | $b_{32}^{+}$ |       |
| Q42           | $b_{22}^{-}$ |       |
| Q43           | $b_{12}^{-}$ |       |
| Q44           | $b_{32}^{-}$ |       |
| P47           | $b_{33}^{+}$ |       |
| P48           | $b_{23}^{+}$ |       |
| P49           | $b_{13}^{+}$ |       |
| Q47           | $b_{33}^{-}$ |       |
| Q48           | $b_{23}^{-}$ |       |
| Q49           | $b_{13}^{-}$ |       |

# POTENTIOMETER SET-UP (MISALIGNMENT)

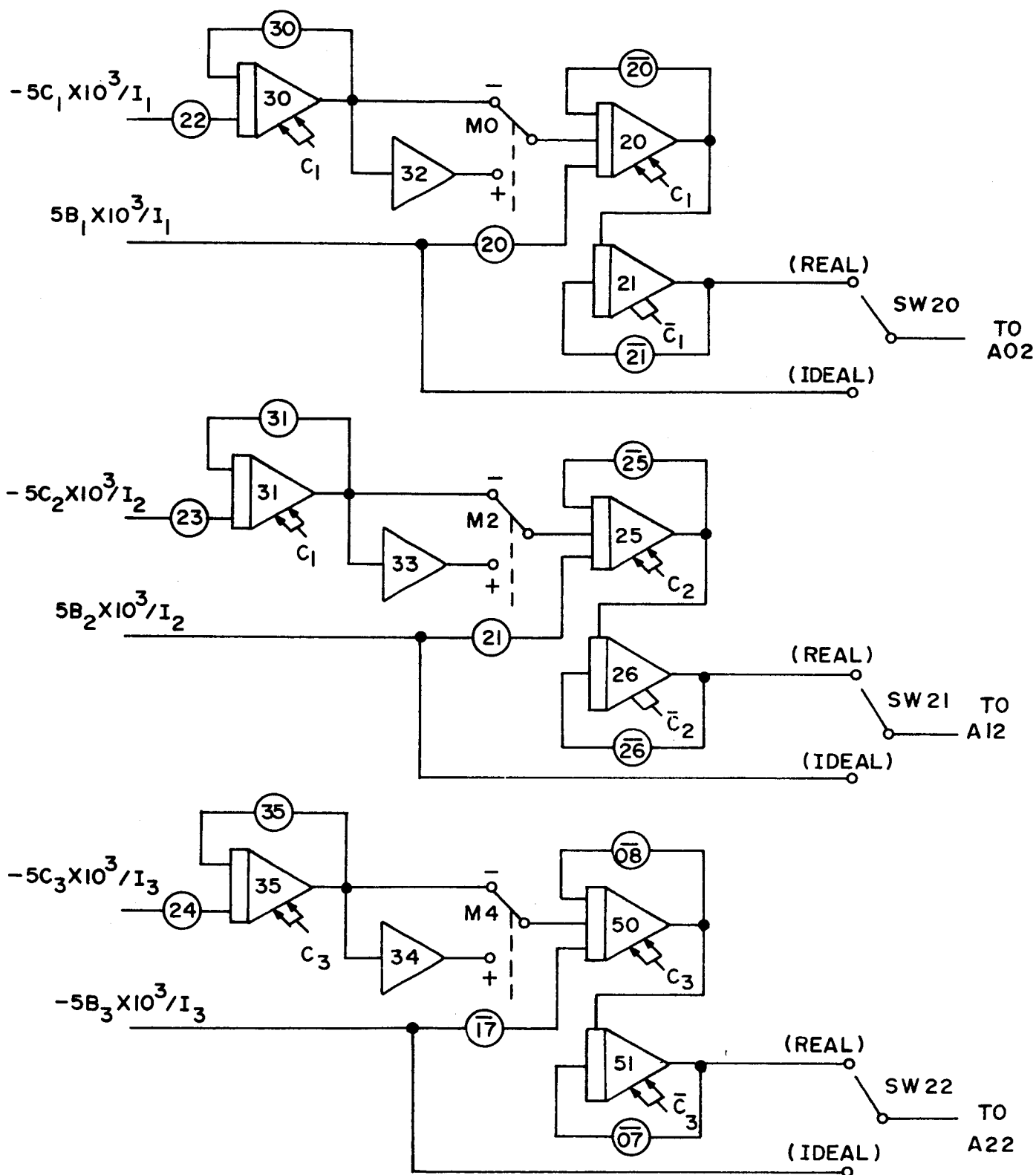


# THRUSTER MISALIGNMENT

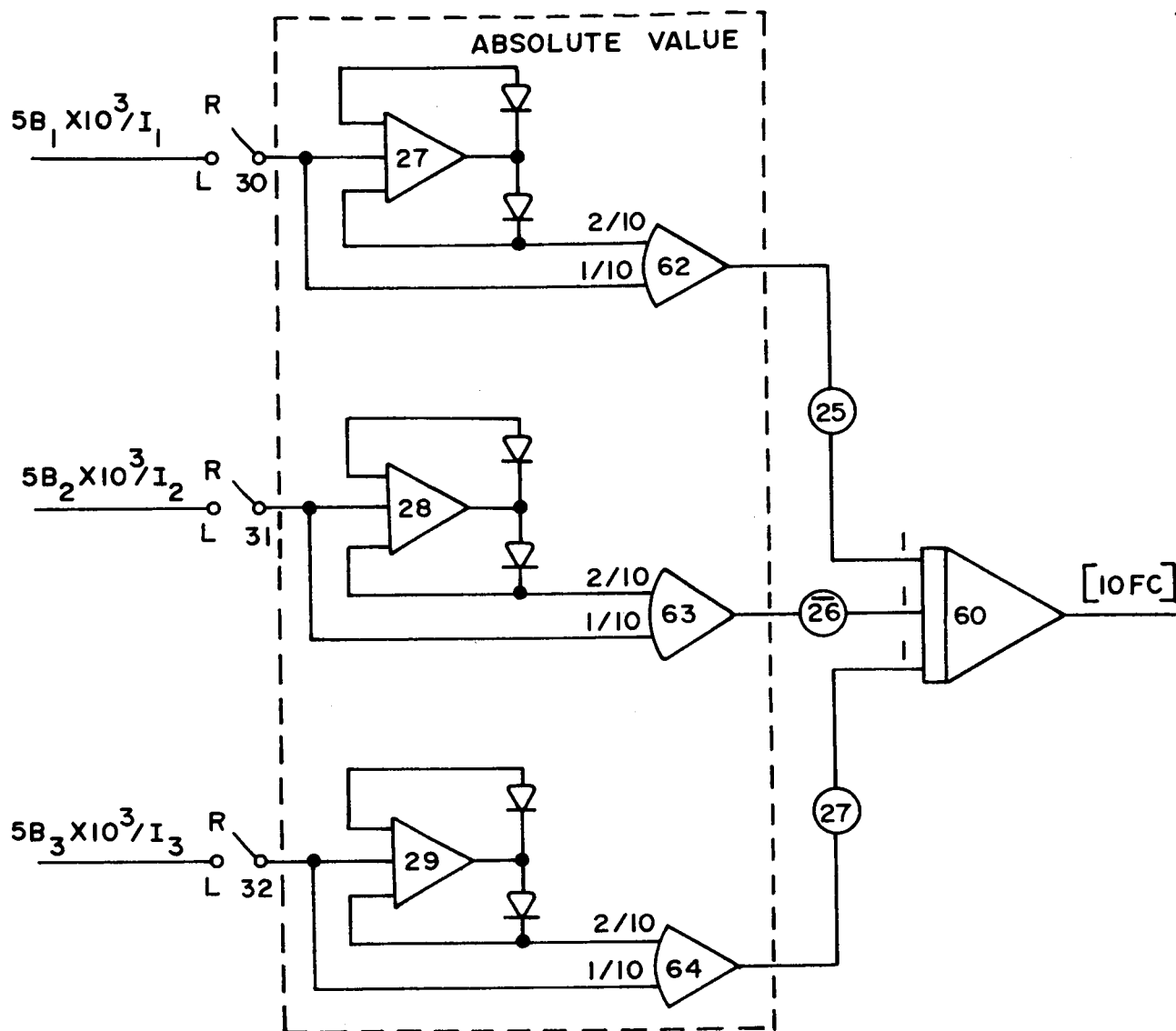
| SWITCH | POSITION      |               |
|--------|---------------|---------------|
|        | RIGHT         | LEFT          |
| 12     | $-b_{21}^{+}$ | $+b_{21}^{+}$ |
| 13     | $-b_{21}^{-}$ | $+b_{21}^{-}$ |
| 22     | $-b_{31}^{+}$ | $+b_{31}^{+}$ |
| 23     | $-b_{31}^{-}$ | $+b_{31}^{-}$ |
| 30     | $-b_{12}^{+}$ | $+b_{12}^{+}$ |
| 31     | $-b_{12}^{-}$ | $+b_{12}^{-}$ |
| 32     | $-b_{32}^{+}$ | $+b_{32}^{+}$ |
| 33     | $-b_{32}^{-}$ | $+b_{32}^{-}$ |
| 40     | $-b_{23}^{+}$ | $+b_{23}^{+}$ |
| 41     | $-b_{23}^{-}$ | $+b_{23}^{-}$ |
| 42     | $-b_{13}^{+}$ | $+b_{13}^{+}$ |
| 43     | $-b_{13}^{-}$ | $+b_{13}^{-}$ |

SWITCH SETTINGS  
(THRUSTOR MISALIGNMENT)

THRUSTOR REAL EFFECTS & FUEL CONSUMPTION



# THRUSTER REAL EFFECTS



$$Q25 = \frac{100 I_1}{d_1 \times \text{ISP} \times 5 \times 10^4} = .0300$$

$$P26 = \frac{100 I_2}{d_2 \times \text{ISP} \times 5 \times 10^4} = .0210$$

$$Q27 = \frac{100 I_3}{d_3 \times \text{ISP} \times 5 \times 10^4} = .0180$$

FUEL CONSUMPTION

| POTENTIOMETER | VARIABLE                                       | VALUE |
|---------------|------------------------------------------------|-------|
| Q25           | $100I_1/d1 \times I_{sp} \times 5 \times 10^4$ | .0300 |
| P26           | $100I_2/d2 \times I_{sp} \times 5 \times 10^4$ | .0210 |
| Q27           | $100I_3/d3 \times I_{sp} \times 5 \times 10^4$ | .0180 |
| Q20           | $K_{12}$                                       | .2800 |
| Q21           | $K_{22}$                                       | .2800 |
| Q22           | $K_{11}$                                       | .0018 |
| Q23           | $K_{21}$                                       | .0018 |
| Q24           | $K_{31}$                                       | .0018 |
| Q30           | $K_{11}$                                       | .0150 |
| Q31           | $K_{21}$                                       | .0150 |
| Q35           | $K_{31}$                                       | .0150 |
| P08           | $K_{32}$                                       | .2800 |
| P07           | $K_{33}$                                       | .0750 |
| P17           | $K_{32}$                                       | .2800 |
| P20           | $K_{12}$                                       | .2800 |
| P21           | $K_{13}$                                       | .0750 |
| P25           | $K_{22}$                                       | .2800 |
| P26           | $K_{23}$                                       | .0750 |

POT SET-UP  
(TRE AND FUEL CONSUMPTION)

| TRE * \ SWITCH | 20 |   | 21 |   | 22 |   |
|----------------|----|---|----|---|----|---|
|                | R  | L | R  | L | R  | L |
| YES (AXIS 1)   |    | X |    |   |    |   |
| NO (AXIS 1)    | X  |   |    |   |    |   |
| YES (AXIS 2)   |    |   |    | X |    |   |
| NO (AXIS 2)    |    |   | X  |   |    |   |
| YES (AXIS 3)   |    |   |    |   |    | X |
| NO (AXIS 3)    |    |   |    |   | X  |   |

\* THRUSTOR REAL EFFECTS

| FUEL CONSUMPTION \ SWITCH | 30 |   | 31 |   | 32 |   |
|---------------------------|----|---|----|---|----|---|
|                           | R  | L | R  | L | R  | L |
| YES (AXIS 1)              |    | X |    |   |    |   |
| NO (AXIS 1)               | X  |   |    |   |    |   |
| YES (AXIS 2)              |    |   |    | X |    |   |
| NO (AXIS 2)               |    |   | X  |   |    |   |
| YES (AXIS 3)              |    |   |    |   |    | X |
| NO (AXIS 3)               |    |   |    |   | X  |   |

SWITCH SETTINGS  
(TRE AND FUEL CONSUMPTION)